2018

Matching: The Search For Control

Colman Hubert Humphrey
University of Pennsylvania, colmanhumphrey@gmail.com

Matching: The Search For Control

Abstract
Matching allows us to estimate the effect of a chosen variable, providing highly interpretable inference without parametric assumptions. When matching, finding good controls is where nearly all the difficulty lies. We develop a theoretical framework and a methodology to generate a set of matches, evaluate them and select a best match given the input variables. We apply this method to a problem of interest, urban data in Philadelphia. In this setting, we also outline our full data collection pipeline in order to encourage replication. In a separate time series setting, we propose a latent model in order to generate probabilities at each time point; these form the basis of an interrupted time series match.

Degree Type
Dissertation

Degree Name
Doctor of Philosophy (PhD)

Graduate Group
Statistics

First Advisor
Shane Jensen

Second Advisor
Dylan Small

Keywords
lottery, mahalanobis, matching, philadelphia, propensity, urbanism

Subject Categories
Mathematics | Statistics and Probability | Urban Studies and Planning

MATCHING: THE SEARCH FOR CONTROL

Colman Humphrey

A DISSERTATION

in

Statistics

For the Graduate Group in Managerial Science and Applied Economics

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2018

Co-Supervisor of Dissertation

Shane Jensen
Associate Professor of Statistics

Co-Supervisor of Dissertation

Dylan Small
Professor of Statistics

Graduate Group Chairperson

Catherine Schrand
Celia Z. Moh Professor, Professor of Accounting

Dissertation Committee

John MacDonald, Professor of Criminology and Sociology
Rachel Thurston, Architect, NCARB

MATCHING: THE SEARCH FOR CONTROL

Dedicated to Amaris and my parents

Acknowledgement

This thesis could not have been made without the support of my advisors, Shane Jensen and Dylan Small. Without me even having to ask, they formed an advising team that lead me to where I am. I would like to thank my co-authors Rachel Thurston and Andrea Troxel. Rachel was invaluable in my urban research, both in terms of knowledge about how exactly city data is collected and what biases are present, and in providing insights and ideas from her background into creating sensible and testable hypotheses. Andrea's domain knowledge expertise in epidemiology and statistical knowledge was invaluable in creating the methods in my lottery research.

I would like to thank all the staff and professors in both the statistics department and in UPenn, and thanks to all my fellow students for the journey.

My parents have always been great supporters of mine in whatever I've done, even from over 5,000 km away, and certainly this PhD was no exception. My wife made the whole process survivable.

Thank you.

Colman

ABSTRACT

MATCHING: THE SEARCH FOR CONTROL

Colman Humphrey

Shane Jensen, Dylan Small

Matching allows us to estimate the effect of a chosen variable, providing highly interpretable inference without parametric assumptions. When matching, finding good controls is where nearly all the difficulty lies. We develop a theoretical framework and a methodology to generate a set of matches, evaluate them and select a best match given the input variables. We apply this method to a problem of interest, urban data in Philadelphia. In this setting, we also outline our full data collection pipeline in order to encourage replication. In a separate time series setting, we propose a latent model in order to generate probabilities at each time point; these form the basis of an interrupted time series match.

Table of Contents

Chapter 1

Introduction

Inference is hard. Causal inference is harder. Even after we make assumptions we feel are reasonable, understanding the causal effect of an input can be tricky.

Black box models are excellent for modelling joint distributions, but are hard to tease apart. Parametric models seem easier, and yet even those of us trained in statistics make misstatements about parametric model inference all the time, not to mention the assumptions required.

Matching can offer insight where other models fail. It is a non-parametric method with a transparent structure and outcome. When units of any kind differ on inputs, we want to attribute the difference in outcomes to this difference in inputs, allowing for some noise. When units only differ on some variable of interest but are otherwise similar, we want to attribute differences in outcome to this specific input variable.

In this thesis, we work through two applied problems, attempting to solve both with various matching methods. In the first of these applied problems, a study of Philadelphia, we develop theory and methodology to introduce a novel matching method in order to answer questions of interest about crime. In the second, a set of medical time series, we combine modern latent process modelling with a mature matching technique to gain insight into the affect of mini lotteries.

The bulk of the theoretical work of this thesis is done in chapter 4. We build on the introduction to matching given in the following section, all the way up to introducing a new procedure in doing matching from scratch for a given problem. For when the full procedure is not necessary for any given problem, we develop practical methods at every stage.

The rest of this introductory chapter introduces the method and assumptions of causal infer-

ence. Our first analyses, in chapter 3, will not use the majority of our novel methodological methods; our second set in chapter 5 will.

1.1 Causal Introduction and Matching

Matching is a popular method in causal inference for estimation. To a large degree, it's exactly as it sounds: matching humans, items, places, things, anything. We then take the outcome of such a match, and produce better analyses than we can with unmatched data.

One of the main outcomes of matching is to create matched sets containing treated and control units. This includes sets of size two: creating matched pairs. Analysis is then done on the level of pairs and potentially combined. A natural example of this are studies on twins.

The other main use of matching is to create just two sets, a treated set and a control set, such that they look like each other. This is often referred to as pruning. Analysis is then performed on this subset of the full data, reducing worries about extrapolation.

Matching can be an efficient non-parametric estimation technique, avoiding model dependence. It provides control, and reduces confounding issues.

Further, matching can provide insight into an observational study, by allowing researchers to analyse differences between treated and control units on background variables, see for example Rosenbaum's Thick Description (Rosenbaum and Silber 2001). When we analyse the coefficient of a variable in a regression framework, we often say the coeffficient represents the effect of the variable adjusting or controlling for all others. Apart from wanting to avoid parametric assumptions of models, matching also much more literally tries to control for the other variables. For example, we can look at typical matched pairs from our analysis and judge qualitatively if they really do seem like appropriate matches - do they really seem close in covariates, do they seem like they could potentially differ on any unmeasured variables?

Matching is not a causal panacea: a poor choice of variables to match on will create a poor causal estimate, no matter how good the matching procedure.

1.1.1 Notation and Assumptions

We assume a typical notational setup for the triple (Y, X, T), coming from some joint distribution. That is, we have a population of units[1] of some kind, with an outcome of interest Y, a covariate of particular interest T and a vector of covariates not generally of primary interest X. X is commonly multivariate, while Y and T are typically univariate. In fact, T is normally binary.

1.1.2 Targets of Causal Inference

Ignoring Y, X and T have some joint distribution in our data. We'll let $f_t(\mathbf{x}) = f(\mathbf{x} \mid t),\ t \in \{0, 1\}$, the conditional densities. The unconditional density of $\mathbf{x}$ is just a weighted average of f_1 and f_0, weighted by the overall probability of treatment and control, a typical marginal density definition.

We define the following objects:

$$\begin{aligned} \mu(\mathbf{x}, t) &= \mathbf{E}\big[Y \big| X = \mathbf{x}, T = t\big] \\ \sigma^2(\mathbf{x}, t) &= \mathrm{Var}\big[Y \big| X = \mathbf{x}, T = t\big] \\ \tau(\mathbf{x}) &= \mu(\mathbf{x}, 1) - \mu(\mathbf{x}, 0) \end{aligned} \tag{1.1}$$

Let Y_1 be the outcome for a unit if that unit had been treated, and Y_0 the outcome if not. Let T be the treatment status for a unit, such that the actual outcome Y is $TY_1 + (1-T)Y_0$. Thus if a unit has $T = 1$, then $Y = Y_1$, and Y_0 is the counterfactual. Really we can only write this in this way if we assume stable unit treatment values, detailed in section 1.1.2 below. Let X be the covariates.

[1] Units can be any unit of inference. In our urban analyses, units are locations in Philadelphia. In our lottery analysis, units are people, or participants.

The average treatment effect is:

$$\text{ATE} = \mathbf{E}[Y_1 - Y_0] \tag{1.2}$$

That is, the expected difference between what would happen under treated and under control.

The average treatment effect for the treated conditions on treated units:

$$\text{ATT} = \mathbf{E}[Y_1 - Y_0 | T = 1] \tag{1.3}$$

This is often of interest because we want to know how a treatment effects units likely to actually receive it: we want to know if a drug helps those who would take it, even if it has mild negative side effects for those who would not.

Of course we can change $f_1(\mathbf{x})$ in many ways: an experimental drug might be given to the most extreme cases initially; once it's proven to be effective, it can be given more generally. A policy proposal might be initially tested where it's thought to be most effective, and subsequently applied more generally.

A more nebulous concept is the Feasible Average Treatment Effect for the Treated, or FATT: it's the average treatment effect for the treated on a subset where comparisons are possible. We'll discuss this further below.

Strong Ignorability

The most vital assumption when matching is on the covariates. We assume they provide strong ignorability. That is, the potential outcomes for a given unit Y_1, Y_0, i.e. what would happen if that unit received the treatment or did not, is independent of treatment status,

given the covariates.

$$Y_0, Y_1 \perp\!\!\!\perp T \,\Big|\, X \tag{1.4}$$

That is, for any set of covariates, loosely speaking any "type" of units, knowing what would happen to them under the two conditions doesn't tell you what treatment status they actually have.

This is equivalent to Pearl's back-door criterion, defined in Pearl et al. (2009).

Let's work through a counter example. Say we want to measure the effect of teaching people how to program, on health. Let $T = 1$ if a person knows how to program. Say that knowing how to program means you can get a better salary, and a better salary means better health through any number of means[2], and no other pathways exist. The true causal effect of the treament is therefore positive, and the pathway to the effect is through salary.

If we condition on salary, we won't have strong ignorability. If we somehow knew the counterfactual health for the two programming outcomes, we would learn information about the treatment: if $Y_1 \gg Y_0$ for a high salary individual, then this implies programming would help this person, which means they were more likely to have learned programming, given we see they have a high salary; more likely than the average high salary individual. For a low salary individual, the opposite occurs: if programming would help them, and they have a low salary, they probably didn't learn to program.

The idea is that for most people, programming affects salary, and salary affects health: if programming does not affect their health, it likely does not affect their salary, therefore we learn little about their programming status from knowing their salary; however the average high salary *does* increasing programming status[3].

[2] E.g. better insurance

[3] On the flip side, if we have $Y_0 \approx Y_1 =$ Good Health for a high salary individual, this implies a lower probablity of $T = 1$ than for all high salary individuals: in this case, programming didn't seem to help their health, which means programming was less likely to affect their job, which means they were less likely to have been taught to program than the average high salary individual, since they have a high salary either

Overlap

We have overlap when we find both treated and control units at all levels of covariates.

The required strength of assumption depends on our target of inference. If we are interested in the ATE, we need the probability of treatment to be bounded away from both zero and one. That is, we have some $\varepsilon > 0$ such that:

$$\varepsilon < \mathbf{P}(T = 1 \mid X) < 1 - \varepsilon \quad \forall\, X \tag{1.5}$$

If we are interested in the ATT, we need only:

$$\mathbf{P}(T = 1 \mid X) < 1 - \varepsilon \quad \forall\, X \tag{1.6}$$

By Bayes' theorem, these overlap principles are mostly identical to assuming we have overlap in terms of $f_1(\mathbf{x})$. For details and relaxations, see Heckman, Ichimura, and Todd (1998).

Stable Unit Treatment Value Assumption

Commonly referred to as just SUTVA, this assumption is often implictly assumed in causal analysis. The extent of this assumption depends on the problem, but generally we assume there are no interaction treatment effects: the response Y of a unit depends only on the treatment for that unit, and not on the treatment assignments of the others.

In fact, SUTVA violations do not ruin all possible versions of the average treatment effect, it just means we can no longer consider pure counterfactuals such as $Y_1 - Y_0$ well defined. Our target of inference could instead be the difference in expected value of a unit under the two assignments, averaged over all possible assignments for other units. Unsurprisingly this is much harder to analyse, and only becomes easier with a different set of assumptions.

way. In fact the same process shows that a low salary individual with good health either way would be less likely to have learned than the average low salary individual.

Distributional Definition of Treatment Effects

Under the three assumptions above, using the defintion of conditional expectation (Kolmogoroff 1933) we can write $\tau = \mathbf{E}\big[\tau(\mathbf{x})\big]$ as the average treatment effect, averaged over the distribution of $\mathbf{x}$. That is:

$$\tau = \int_{\mathbf{x}} f(\mathbf{x})\tau(\mathbf{x})\ \mathrm{d}\mathbf{x} \tag{1.7}$$

Once it's clear we've made the causal assumptions, we'll use τ and ATE interchangeably.

Similarly, $\tau_t = \mathbf{E}\big[\tau(\mathbf{x})\big|T = 1\big]$ is the average effect of the treatment on the treated: now we average over the conditional distribution of x when $T = 1$, $f_1(\mathbf{x})$:

$$\tau_t = \int_{\mathbf{x}} f(\mathbf{x} \mid T = 1)\tau(\mathbf{x})\ \mathrm{d}\mathbf{x} = \int_{\mathbf{x}} f_1(\mathbf{x})\tau(\mathbf{x})\ \mathrm{d}\mathbf{x} \tag{1.8}$$

Again, we'll use ATT and τ_t interchangeably.

When we don't have overlap, all is not lost. We can simply change our target of inference to the previous mentioned FATT, the Feasible ATT. In fact, this is already what τ_t is, the overlap assumption is what allows τ_t to be the ATT. Essentially we can never know the treatment effect for units that have zero probability of receiving the treatment, and we likely don't care, so we focus on estimating the effect on units that the treatment could affect. This is not altogether hugely different from moving from ATE to ATT.

For a given unit i, let $\varepsilon_i = Y_i - \mu(\mathbf{x}, T_i)$, the residual.

1.1.3 Notation for Bipartite Matching

A bipartite match is where we have two distinct groups, often Treated and Control.

Assume we have N units. Let $T_i = 1$ if unit i is treated, $T_i = 0$ if unit i is a control, with $\mathbf{T}$ being the full vector. We let $N_t = \sum_{i=1}^{N} T_i$ be the number of treated units, and $N_c = N - N_t$ the number of controls. We don't need more controls than treated units, i.e. $N_c > N_t$, but it is commonly true.

Let $\mathbf{x}_i$ be the vector of covariates for unit i, and $\mathbf{X}$ the matrix of all covariates, such that the ith row of $\mathbf{X}$ is $\mathbf{x}_i'$. Let $\mathbf{x}^{(j)}$ be the jth column of $\mathbf{X}$, i.e. it's the vector of the jth covariate.

1.1.4 Notation for Non-Bipartite Matching

In different contexts, non-bipartite matching, or NBP, can mean different things. For us, a non-bipartite match is any match where we don't have two distinct groups, i.e. the "treatment" variable of interest can take on multiple values, or even be multivariate. Our assumptions about X and Y are relatively unchanged, but now the conterfactuals are defined over all values of the treatment, Y_t, and thus overlap and ignorability generalise too.

The first two objects defined in equation 1.1 remained unchanged, but indeed we no longer have a well-defined value of $\tau(\mathbf{x})$. The most general NBP target of inference is:

$$\tau(\mathbf{x}, t_a, t_b) = \mu(\mathbf{x}, t_a) - \mu(\mathbf{x}, t_b) \tag{1.9}$$

We might focus on identifying the treatment effect for a specified difference:

$$\tau(\mathbf{x}, \delta) = \mathbf{E}_\nu\big[\mu(\mathbf{x}, \nu) - \mu(\mathbf{x}, \nu - \delta)\big] \tag{1.10}$$

And of course we can define $\tau(\delta)$ as the expected value of $\tau(\mathbf{x}, \delta)$ over $\mathbf{x}$.

We could also define the "derivative" of $\tau(\mathbf{x}, \delta)$ with respect to delta:

$$\mathbf{E}_{\nu,\delta}\left[\frac{\mu(\mathbf{x}, \nu) - \mu(\mathbf{x}, \nu - \delta)}{\delta}\right] \tag{1.11}$$

Averaging the above object over $\mathbf{x}$, while looking complicated, is a reasonable target of NBP matching under certain assumptions. If the treatment effect is linear, we'll have $\mu(\mathbf{x}, t) = \mu(x) + \beta t$. Then $\frac{\mu(\mathbf{x},\nu)-\mu(\mathbf{x},\nu-\delta)}{\delta} = \beta$.

1.1.5 Randomised Controlled Trials

In randomised controlled trials, or RCTs, we typically randomise the treatment assignment. This breaks the association between T and Y, X, thus we get ignorability automatically.

Nearly all RCT strategies force $f_1(\mathbf{x}) = f_0(\mathbf{x})$[4], thus we get overlap for free.

Using the same techniques as section 4.5, under randomisation of T we can show that the treated average outcome minus the control average outcome:

$$\frac{1}{N_t}\sum_{i=1}^{N} Y_i T_i - \frac{1}{N_c}\sum_{i=1}^{N} Y_i(1-T_i) \tag{1.12}$$

is an unbiased estimate of the ATE.

1.2 Contributions

All theory, analyses and plots in the current thesis are my own, and I am responsible for any and all errors.

The writing in sections 2.1, the introduction to urban vibrancy, and in chapter 3 were contributed to in large part by Shane Jensen, along with Dylan Small, Rachel Thurston and myself. Most of this work will appear in *Analysis of Urban Vibrancy in Philadelphia*, under review.

Sections 6.1 and 6.3 were written with great help from Andrea Troxel[5] and Dylan Small. Most of this work will appear in *Modelling Lottery Incentives on Daily Adherence*[6].

[4] Of course we also can control $f(\mathbf{x})$ by deciding who is part of the study.
[5] Professor, Department of Population Health, NYU
[6] Working title

Chapter 2

Urban Planning and Data for Philadelphia

In this chapter, we introduce the motivation and history behind studying urban environments, and specifically why we study Philadelphia. We further introduce and describe the data used in our urban analyses, in chapters 3 and 5.

Core urban data refers to data used in both sets of analyses, although at different resolutions. We detail this data first in the context of the analyses in chapter 3.

Augmented urban data refers to data used only in the second set of analyses, in chapter 5. We also detail modifications of the core data for our intersection level analyses.

We defer the description of data from our lottery project to 6, for narrative flow and consistency.

2.1 An Introduction to Urban Vibrancy and Safety in Philadelphia

Throughout history there have been many perspectives on the approach to planning of cities, with a notable clash between dense, organically-formed urban spaces versus large-scale clearing and planning of "superblocks" and automobile-centric layouts. The former perspective viewed city development as a social enterprise created by many hands, whereas the top-down central planning approach involved less input from the residents affected by city changes. The urban renewal movement of the 1960s and 1970s is the largest example of this latter effort, but the same mentality still drives many current development decisions.

One historical motivation for top-down urban renewal projects was the idea that cities were over-crowded. Winsborough (1965) discusses both positive and negative perspectives on the effects of population density in urban settings. Population density has been positively asso-

ciated with division of labor but has also been linked to psychological strain and negative health outcomes. Simmel (2011) argues that the emotional stress caused by high population density produces negative attitudes and hostility among the populace. In a study of Baltimore, Verbrugge and Taylor (1980) find both positive and negative effects of population density and suggest that population size is a more important factor for attitudes and behavior in urban environments.

Earlier responses to anti-density rhetoric and the challenges of urban living during the industrial age resulted in code regulations, restrictive land use zoning, and sometimes large scale clearing of entire neighborhoods. During the age of urban renewal, dense urban environments were demolished and replaced by trending architectural works, civic monuments and tree lined boulevards built for reducing population density and easing automobile traffic, along with large housing projects for displaced communities. Over time, a large number of these projects failed to attract pedestrian activity and resulted in high crime housing areas.

In her seminal work *The Death and Life of Great American Cities* (1961), Jane Jacobs challenged the analyses of proponents of urban renewal and outlined several alternative hypotheses for sustaining successful urban environments. Many of her ideas were based on her own anecdotal observations of urban residents, but can now be investigated quantitatively using recently available urban data.

Jacobs was a pioneering voice for the concept of urban *vibrancy*, a measure of positive activity or energy in a neighborhood that makes an urban place unique and enjoyable to its residents despite the challenges of urban living. An important term coined by Jane Jacobs was "eyes on the street" which summarized her viewpoint that safer and more vibrant neighborhoods were those that had many people engaging in activities (either commercial or residential) on the street level at different times of the day (Jacobs 1961).

This concept of eyes on the street has been more recently expressed as the "natural surveillance" component of the *Crime Prevention through Environmental Design* movement (Deutsch

2016). These contemporary theories argue that the likelihood of criminal activity is strongly linked to the presence or absence of people on the street. As Deutsch (2016) states: "Criminals do not like to be seen or recognized, so they will choose situations where they can hide and easily escape." Policies which promote vibrancy and activity could potentially benefit crime prevention.

The recent explosion in high resolution data on cities offers an exciting opportunity for quantitative evaluation of contrasting urban development perspectives as well as current urban planning efforts. In this paper, we outline a pipeline for data collection and analysis of the associations between neighborhood safety, economic and demographic conditions and the built environment within urban environments.

We target our analysis pipeline towards a more specific goal: using high resolution data to create quantitative measures of the built environment that can serve as proxies for the human *vibrancy* of a local area. We then investigate the association between these vibrancy measures and safety in the city of Philadelphia. We focus on vibrancy proxy measures based on land use as well as business activity, which follows the "natural surveillance" idea that the presence of open businesses encourages safety through the store front presence of both staff and customers.

MacDonald (2015) provides a comprehensive review of past research into the association between the built environment and safety, where many quasi-experimental studies have shown that changes in housing, zoning and public transit can help to manage crime. In section 3.3, we will try to emulate a quasi-experimental setting in our own analysis by comparing locations within census block groups, thereby matching locations in terms of economic health and population density.

The effects of natural surveillance on neighborhood vibrancy can be both subtle and complicated. The presence of a commercial business can encourage vibrancy through the presence of many people coming and going, or can give a sense of vacancy and isolation to an area if it is closed during a particular time of the day. In order to get an accurate picture of

whether commercial businesses help to encourage safety, we will need to examine whether or not those businesses are open and active, as we outline in section 3.2.

We choose the city of Philadelphia as a case study for this work as Philadelphia is currently encountering many contemporary issues in urban revival, population growth and desirability. Recent work has shown that urban city centers are growing relative to their suburban counterparts in many areas of the country (Couture and Handbury 2015). Another study by Ellen, Mertens Horn, and Reed (2017) finds an association between population movement of high-income and college-educated households and declining crime rates in central city neighborhoods.

We first outline our data collection for the city of Philadelphia in section 2.2 and then explore the associations between safety and several economic, population and land use measures in section 3.1. To get a more detailed picture of neighborhood vibrancy, we compile a database and several measures of business vibrancy in section 3.2. In section 3.3, we employ several matching analyses to evaluate the association between business vibrancy and safety within local neighborhoods, and then conclude with a brief discussion in section 3.4.

In order to encourage replication of our urban analyses and adaptation to other research questions, we have made the code and public data that were used in our analyses available as a github repository at:

`https://github.com/ColmanHumphrey/urbananalytics`

2.2 Core Urban Data in Philadelphia

Our analysis in chapter 3 will be based on the geographical units defined by the US Census Bureau. Philadelphia county is divided into 384 census tracts which are divided into 1,336 block groups which are divided into 18,872 blocks. Figure 2.1 (left) gives a map outlining the 1,336 block groups in Philadelphia. Population and economic data are provided by the US Census Bureau, crime data is provided by the Philadelphia Police Department and land use data is provided by the City of Philadelphia.

A general theme of our urban work is that results can vary (often substantially) depending on the resolution level of the data and the geographic scale of the underlying processes involved. Most of our analyses will be done at the block group level which allows for the best interoperability between our economic and built environment data, but we also perform several analyses at the block level.

We use shape files provided by the US Census Bureau for our population and economic data and shape files provided by the City of Philadelphia for the land use data. Shape files from the Census Bureau delineate the boundaries and area of each census block and block group. Shape files from the City of Philadelphia delineate the boundaries and area of each lot in Philadelphia. For the vast majority of lots in Philadelphia, the lot is entirely contained within a single Census Bureau block. We outline further details of each data source below.

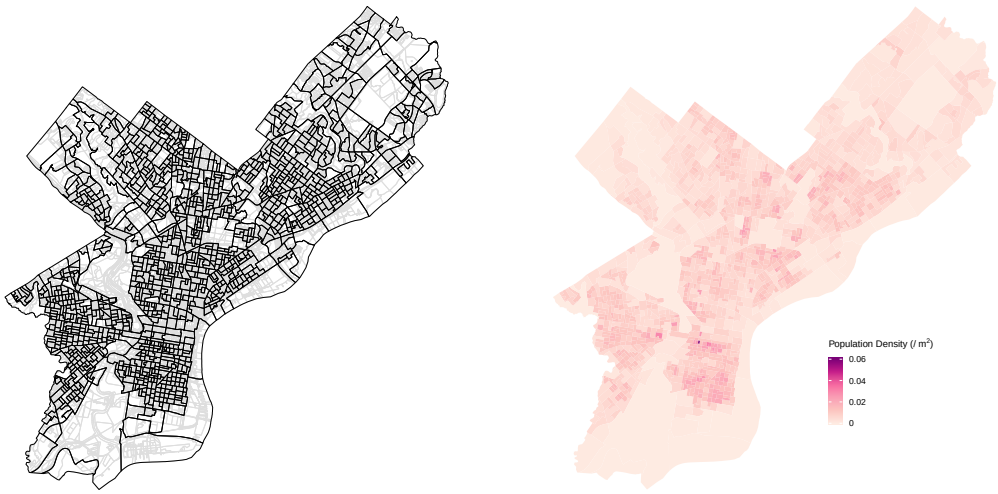


Figure 2.1: **Left:** Map of Philadelphia divided into block groups (black lines) by US Census Bureau. **Right:** Population density by block group in Philadelphia

2.2.1 Population Data

Our population demographic data were pulled from the census website (`factfinder.census.gov`) by setting the geography as all blocks in Philadelphia and setting the data source as "Hispanic or Latino Origin By Race" (which is SF1 P5 in their database). The raw demographic

data gives the population count in each block from the 2010 census.

Of the 18,872 blocks in Philadelphia, 4,558 have no residents (e.g. parks, industrial areas, etc.). At the block level, we restrict our analysis to to blocks with at least 25 people, which gives 12,874 blocks that contain 98.9% of the population. At the block group level, we restrict our analysis to block groups with at least 400 people in them. which gives 1,325 block groups (out of 1,336) that contain 99.96% of the population.

We calculate the population count and population density in each block group i from the raw population data and using the area of each block group from the US Census Bureau shape files. Figure 2.1 (right) gives the spatial distribution of the population density across Philadelphia.

2.2.2 Economic Data

Our economic data were pulled from the American Community Survey on the census website (`factfinder.census.gov`): tables B19301 for income and C17002 for poverty, both from 2015. This data is only available at the block group resolution level.

For each block group in Philadelphia, we have the per-capita income and the fraction of the population in seven different brackets of income-to-poverty-line ratios: $[0, 0.5)$, $[0.5, 1)$, $[1, 1.25)$, $[1.25, 1.5)$, $[1.5, 1.85)$, $[1.85, 2)$, $[2, \infty)$. For interpretation, the $[0.5, 1)$ bracket represents families that have income between 50% of the poverty line and the poverty line.

The poverty line threshold for each household is defined by the Census Bureau according to the size and composition of the household. As an example, a family of four with two children has a poverty line threshold of $23,999.

We define a scalar poverty measure for each block group based on the weighted sum of the proportion of block group households in each of the seven poverty brackets:

$$\text{Poverty}_i = \sum_{q=1}^{7} w_q \, p_{i,q}$$

where $p_{i,1}$ is the proportion of block group i households in the lowest bracket $[0, 0.5)$ and $p_{i,7}$ is the proportion of block group i households in the highest bracket $[2, \infty)$. We employ linearly decreasing weights $\mathbf{w} = [1, 5/6, 4/6, 3/6, 2/6, 1/6, 0]$ to give highest weight to the brackets with highest poverty. Our Poverty_i metric varies from 0 to 1, with larger values implying higher poverty: a block group with every household in the $[2, \infty)$ bracket takes the value zero, and one with every household in the $[0, 0.5)$ bracket takes the value one.

Figure 2.2 gives the spatial distribution of income (left) and our poverty metric (right) at the block group level in Philadelphia. We see that the areas of West Philadelphia and North Philadelphia have the lowest incomes and highest levels of poverty.

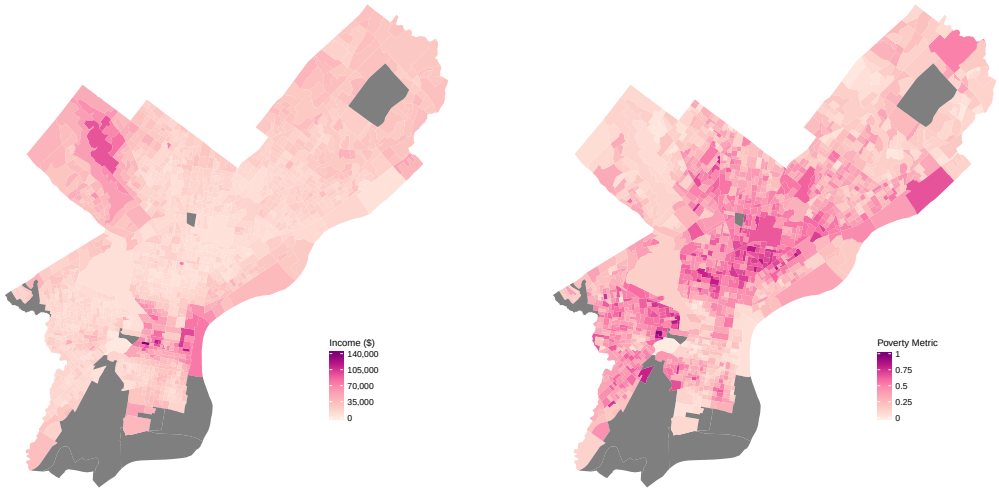


Figure 2.2: **Left:** Per-capita income **Right:** Poverty metric. Block groups that are colored grey do not have enough residents for the US Census Bureau to report economic data for those block groups.

2.2.3 Land Use Zoning Data

Land use zoning data were downloaded from the City of Philadelphia. The land use data consists of a shapefile that divides the city into approximately 560,000 lots and provides the area and registered land use zoning designation (commercial, residential, industrial, vacant, transportation, water, park, civic, recreation, culture, and cemetery) for each of these lots.

As an example, we show the land use designations for overall Philadelphia and the center city neighborhood in Figure 2.3.

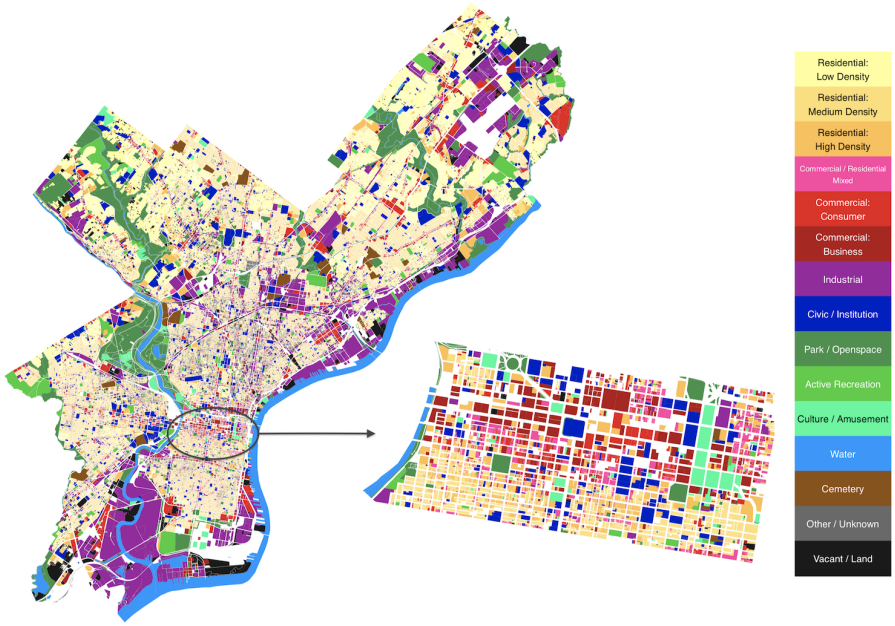


Figure 2.3: Land use designations for overall Philadelphia and the center city neighborhood

Note that we combine the "commercial business" and "commercial consumer" into a single *commercial* designation, and all three "residential" densities into a single *residential* designation. For the rest of this document, *Mixed Use* refers to the designation of "commercial / residential mixed".

We create several land use measures from these zoning designations. First, we calculate the fraction of area in each geographic unit (either block or block group) i that is designated as 'Vacant',

$$\text{Vacant.Prop}_i = \frac{Area_i(\text{Vacant})}{Area_i}$$

Second, we calculate the ratio of the area in each geographic unit (either block or block group) i that is commercial versus residential,

$$\text{ComRes.Prop}_i = \frac{Area_i(\text{Commercial})}{Area_i(\text{Commercial}) + Area_i(\text{Residential})}$$

Finally, we calculate a mixed use proportion, i.e. the proportion of every block or block group that is designated as mixed use,

$$\text{MixedUse.Prop}_i = \frac{Area_i(\text{Mixed Use})}{Area_i}$$

These land use zoning metrics provide our first set of proxy measures for the vibrancy of a local neighborhood. Figure 2.4 gives the spatial distribution of vacant proportion (left) and mixed use proportion (right) at the block group level in Philadelphia.

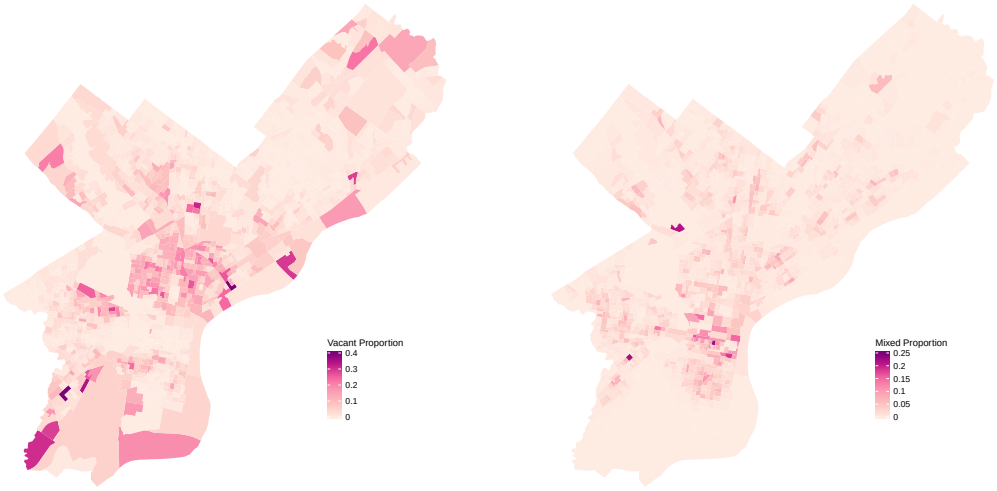


Figure 2.4: **Left:** Vacant Proportion **Right:** Mixed Use Proportion

Philadelphia's zoning procedures were revised in 2012 (http://www.phila.gov/li/Pages/Zoning.aspx). Our zoning data was downloaded in June 2014, and all of our analyses are based on that snapshot. While most of the city's zoning remains unchanged, lots can be rezoned through applications on a continuous basis.

2.2.4 Crime Data

Crime data for Philadelphia comes from the Philadelphia Police Department through the opendataphilly.org website. From their documentation: *Data comes directly from the Police Departments mainframe INCT system. The INCT system is fed by field incident reports*

and Computer Aided Dispatch system. Our dataset consists of all crimes reported by the police in the city of Philadelphia from January 1, 2006 to December 31, 2015.

For each crime, we have the type of crime, the date and time of the crime, and the location of the crime in terms of latitude and longitude (WGS84 decimal degrees). Each crime in our dataset is categorized into one of several types that are listed along with the relative frequency of those types in Figure 2.5. Note that these crime categories are roughly ordered in terms of severity, and that high severity crimes are much less frequent.

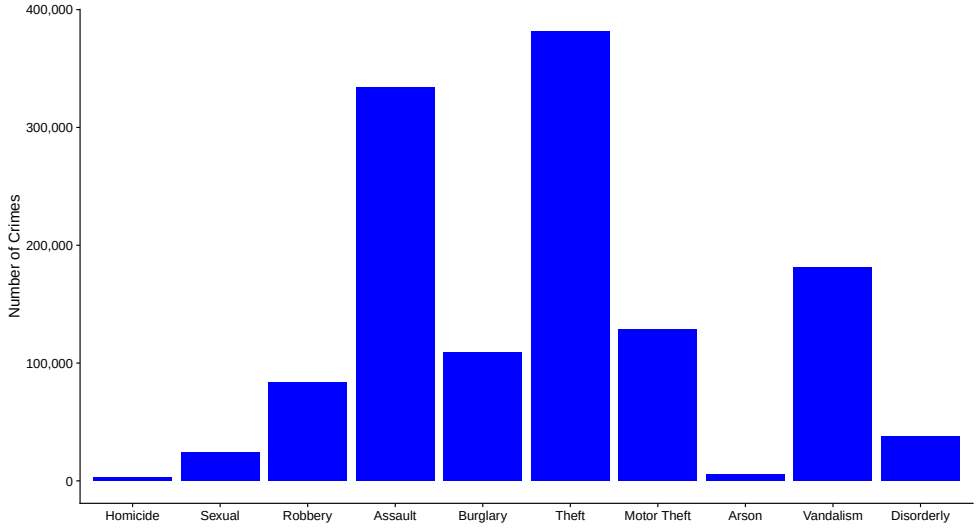


Figure 2.5: Relative frequency of different crime types reported in Philadelphia from January 1, 2006 to December 31, 2015.

For our subsequent analysis, we will combine these categories into two super-categories of crimes: a. **violent crimes** (Homicides, Sexual, Robbery and Assault) and b. **non-violent crimes** (Burglary, Theft, Motor Theft, Arson, Vandalism, and Disorderly Conduct).

Figure 2.6 gives the spatial distribution by block group of violent vs. all crimes committed in Philadelphia from 2006-2015. We see substantial heterogeneity in crime counts across the city with a large outlier count of both violent and non-violent crimes in the Market East block group of center city.

We further discuss crime in section 2.3.2.

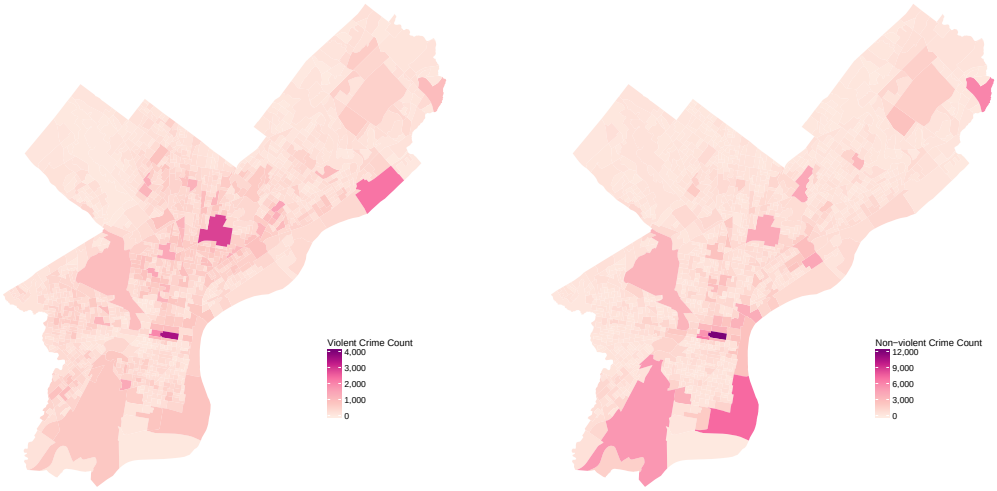


Figure 2.6: Number of Violent and Non-Violent Crimes in Philadelphia: 2006-2015

2.3 Augmented Urban Data in Philadelphia

We outline our new data sources, i.e. data we did not introduce in section 2.2, including how we combine it into intersections. We also detail how we transform our data sources from section 2.2 to match our new data on the intersections level.

2.3.1 Augmented Data

We use intersections as our unit of analysis. We detail in section 2.3.1 how we create these. The main idea is that the intersection of two streets and the surrounding 50 meter radius circle provide us with a useful area of analysis. We avoid the issues associated with block level analyses, where the "action" occurs at the face of the blocks. Further, this small area is more clearly directly affected by the businesses and people in it; measuring one side of a block might be misleading for the opposite site.

Streets and Intersections

Our street data is from `opendataphilly.org`, Street Centerlines dataset. The raw data contains all streets. We remove all streets designated as highways, and form intersections

(a) Philadelphia

(b) Center City

Figure 2.7: Streets and intersections for Philadelphia. Longer streets are darker. Center City on right corresponds roughly to the black box on the left.

from the remaining streets.

We required that all formed intersections are at least 100 meters apart. This means we will not get all intersections, since there are far more intersections one can make in Philadelphia than with this restriction. There are also many possible solutions to such a restriction. We solve this by searching for intersections through streets in order of length. This gives us e.g. intersections formed by connecting South St, Pine St and Spruce St with 19th St, rather than 19th St with Addison, Panama or Cypress.

Figure 2.7 plots the streets and the intersections we formed, both in the whole city and in Center City. The streets are colored according to length, with longer streets darker. The intersections are pink circles.

Overall, this gives us 8,714 intersections in Philadelphia. These form the basis of our matching procedures, and all data is reorganised on the scale of the intersections.

For most of the following data sources, the 50 meter radius circle around each intersection is used to match data. For example, the crime counts around each intersection are the crimes within 50 meters of the centre.

Property Data

Our street data is from `opendataphilly.org`, Property dataset. Around each intersection, we record the number of properties, the average market value, the average age and age deviation[1], the average number of stories, the average number of garages[2], and the price per square foot and its deviation.

Figure 2.8 plots the market value of our properties. Note that the color scale is highly compressed at the top of the range.

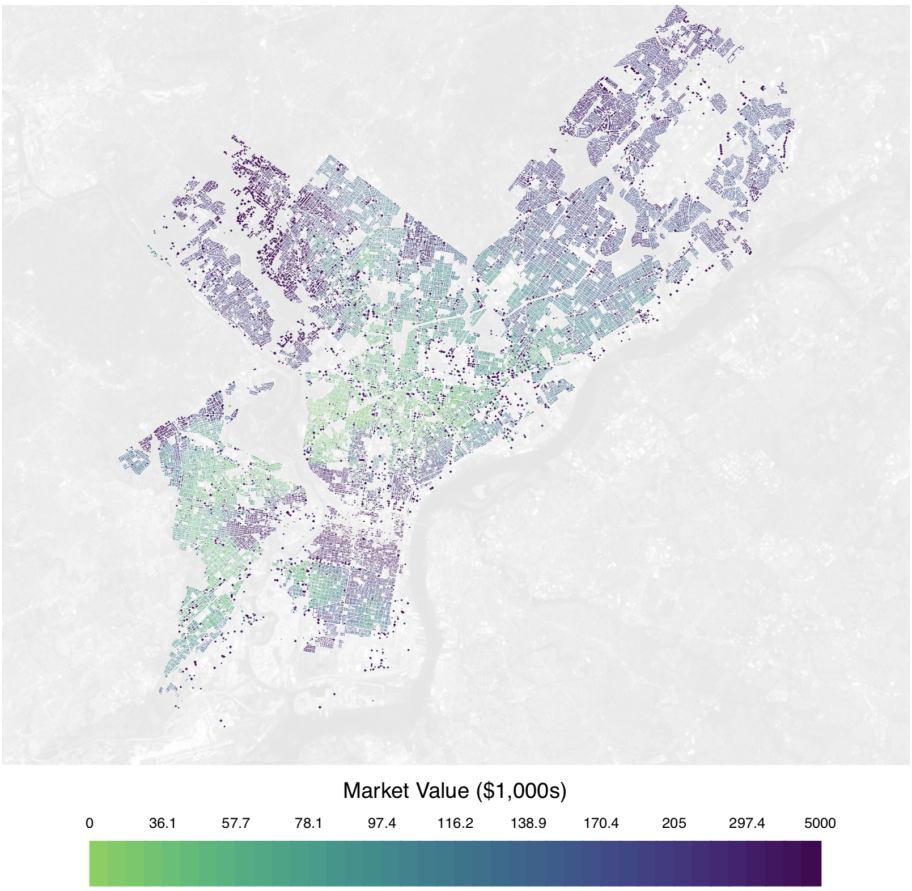


Figure 2.8: Market value of properties in Philadelphia. Area of diamonds is proportional to the area of the properties.

[1]The deviation we use for age is mean absolute deviation from the median. For the price per square foot, we take a weighted average of absolute deviation from the median, weighted by the area of the property.

[2]Generally zero or one, so usually a proportion of properties with garages, but larger buildings have many

School Data

Our school data is from `opendataphilly.org`, Schools dataset. The main variables of interest are enrollment and grades taught in each school. Some schools, mainly private, do not have enrollment data. There are many other sources of data about Philadelphia schools available online, such as serious incident counts, but there are concerns about the bias and validity of the data collection methods. Further, it's possible some of these are counted as crimes in our crime dataset, thus we don't want to match on these to investigate crime.

Figure 2.9 plots the schools we use, with the size of the points proportional to school enrollment the color according to the grades taught. We have data on the type of school: district (public), private, or archdiocese, but we don't use type of school in our analysis, therefore we don't plot it. Small schools, under 150 in enrollment, usually charter schools, are excluded from analysis and not drawn.

For each intersection, we record six variables. The first four are simple: they are the distance to the nearest high school, the distance to the nearest elementary school, and the enrollment at each of these schools[3].

For the last two variables, we compute the density of both High and Elementary school students over the city, with a Gaussian kernel. The bandwidth is selected so that a school's effective count would be half as strong at 500 meters[4]. Each intersection is assigned the total density at its location, summing over all schools.

Schools are an exception to our usual 50 meter rule, in that we use data from beyond the 50 meter radius that defines the intersection.

[3]Schools that are listed as both High and Elementary schools, e.g. schools that teach all grades, can count towards both categories. Thus for a small number of intersections, nearest elementary school and nearest high school are the same.

[4]Thus down to 13.5% at $1,000$ meters

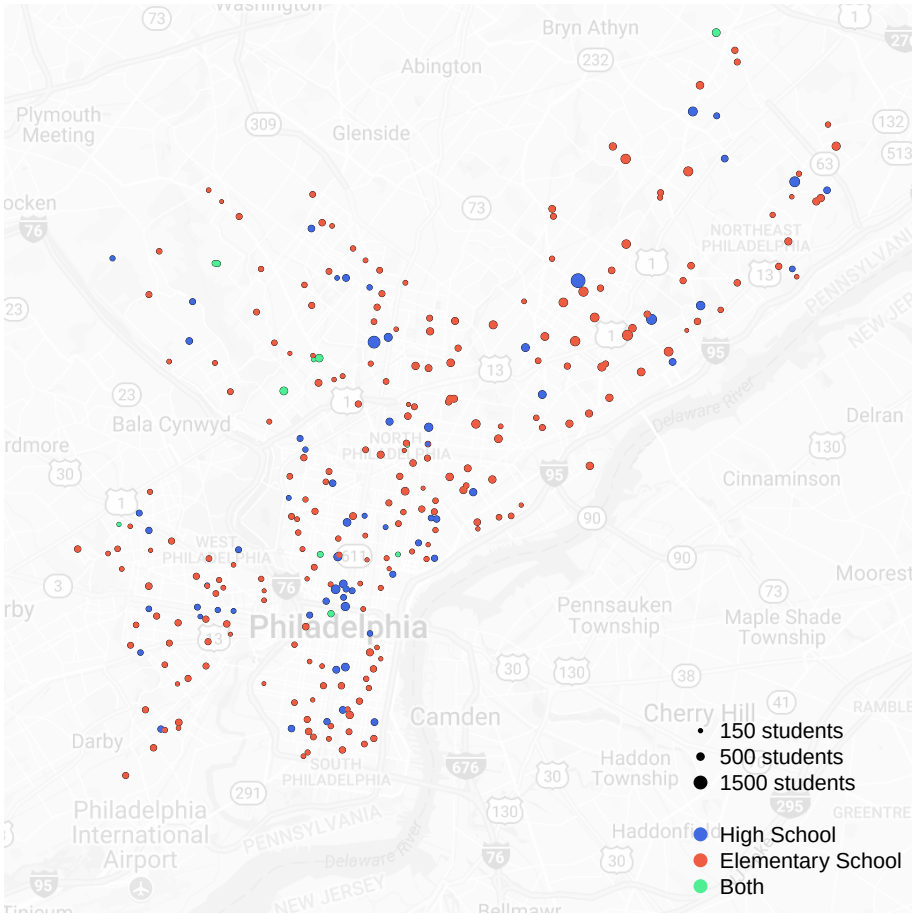


Figure 2.9: Location of each school, along with types and grades. Size of each point is proportional to enrollment.

Transit Data

Our transit data is from `septa.org`, from SEPTA's API. We pull all station / stop data for buses, trolleys and subways. We decided that regional rail stops are not suitable for our analysis.

Figure 2.10 plots the routes. Many routes extend beyond the city limits, but we constrain the plot to Philadelphia for relevance.

Around each intersection, we record the number of routes of each type of transit that have a stop within 50 meters. If for example a bus on a single route stops twice in that radius, we only count it as one; however a single bus stop serving multiple bus routes will count

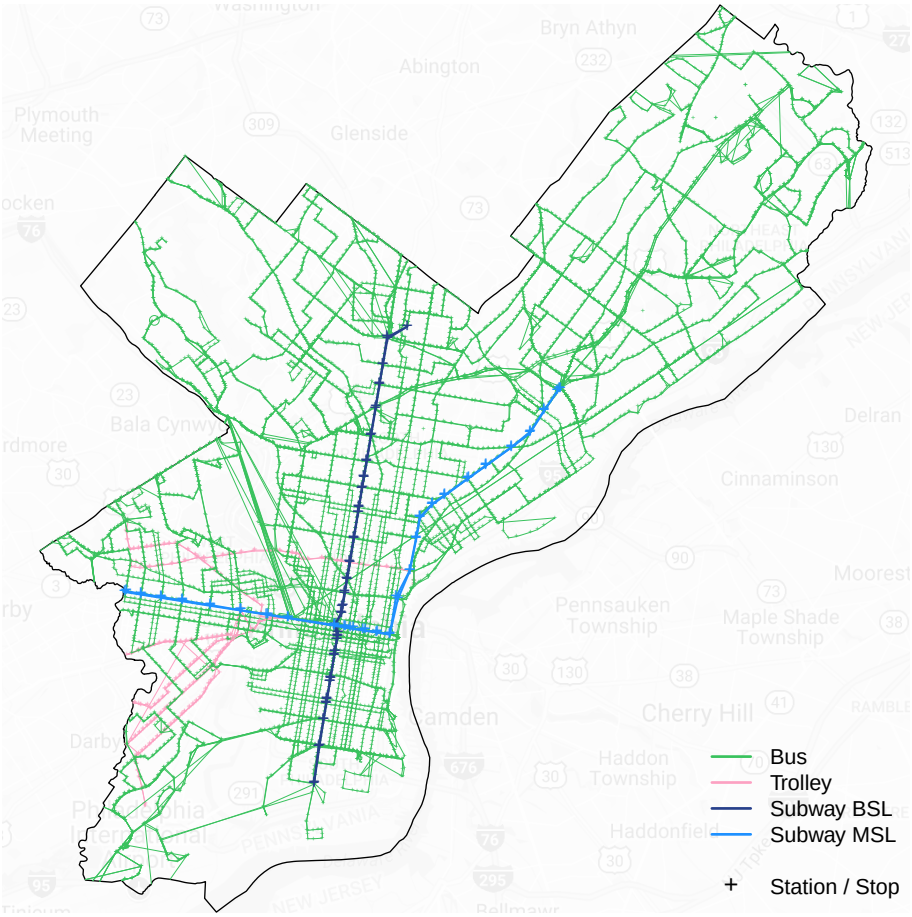


Figure 2.10: Routes for each type of local (SEPTA) public transport, contrained to be within the city.

them all.

Traffic Control Data

Our traffic control data is from `opendataphilly.org`, Intersection Controls dataset. The data contains information on 16,886 intersections in Philadelphia. 3,367 have traffic lights, 3,766 have all way stop signs, and the rest are "conventional". Note that this doesn't account for every overlapping pair of streets, as there is no signal required when the only way onto a street always has right of way[5]. While there is some difference between intersections that have stop signs that are not all-way stop signs, and intersections with no signage at all, we

[5]For example, turning onto an alley, i.e. small one-way, from another one-way - there is no traffic to stop for, except pedestrian.

group them together for our analysis.

Figure 2.11 plots the locations of traffic lights and stop signs in Bella Vista and the surrounding areas.

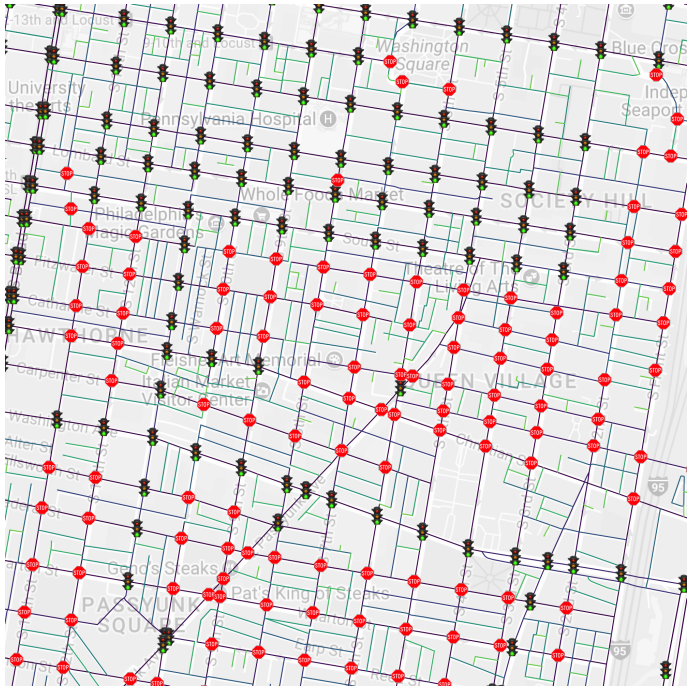


Figure 2.11: Traffic lights and stop signs in Hawthorne, Bella Vista, Queen Village and Society Hill.

Another exception to the 50 meter rule: here, we assign each intersection its closest signal type between the three: traffic lights, all-way stop signs, and all others. If none are close (within 5 meters), we assume there is no stop sign or traffic light. The reason is that we often have signals for other intersections within our circle, if another small intersection is close by.

The location of traffic lights and stop signs is guided by traffic studies, thus we deem traffic lights and all-way stop signs as a useful proxy for the type of traffic at a given intersection.

2.3.2 Core Data in Intersections

We re-use all the data from section 2.2, except landuse data.

Demographic Data

We use our demographic data in two ways. Firstly, we estimate population totals and race proportions within each intersection circle. Secondly, we estimate mean income and our poverty metric[6] in each intersection circle.

We estimate population by scaling the populations in the surrounding blocks by the proportion of their area within each intersection.

For example, if a circle fully contains a given block, the proportion for that block will be 1. If a blockgroup fully contains a circle, our proportion will be $50^2\pi m^2$ divided by the area of the blockgroup. A typical circle will intersect four blocks, and between one and four blockgroups.

Figure 2.12 shows an example for one of our intersection circles, at Walnut St. and 21st St. The blocks $346, 347, 348, 395$ and 400 contain respectively $7, 8, 16, 51$ and 104 people classified as Asian according to the 2010 census. Our circle contains 15.8% of block 400, thus we estimate this sector contains approximately 16.4 Asians; similarly the first four blocks give us $2.4, 0.04$, 3.1 and 7.3 Asians, for a total estimate of 29.9.

We do the same for our other four race categories. Adding these gives our estimated total, and thus we get estimated proportions.

Note that in Figure 2.12, our circle barely intersects block 347. This is reflected in the tiny proportion of that block measured to be in our circle, and indeed the population of that block is barely counted in our circle. In contrast, our circle contains more than $1/3$ of block 346.

Getting income and poverty metrics for each intersection is similar, except on the blockgroup scale. We do the same as the above, using the areas to estimate the population in each connected blockgroup. We then take a weighted average of the income and poverty metric

[6] Essentially a metric that goes from zero in neighborhoods with no poverty, up to one in neighborhoods with all households in the lowest poverty bracket. See prior paper for details.

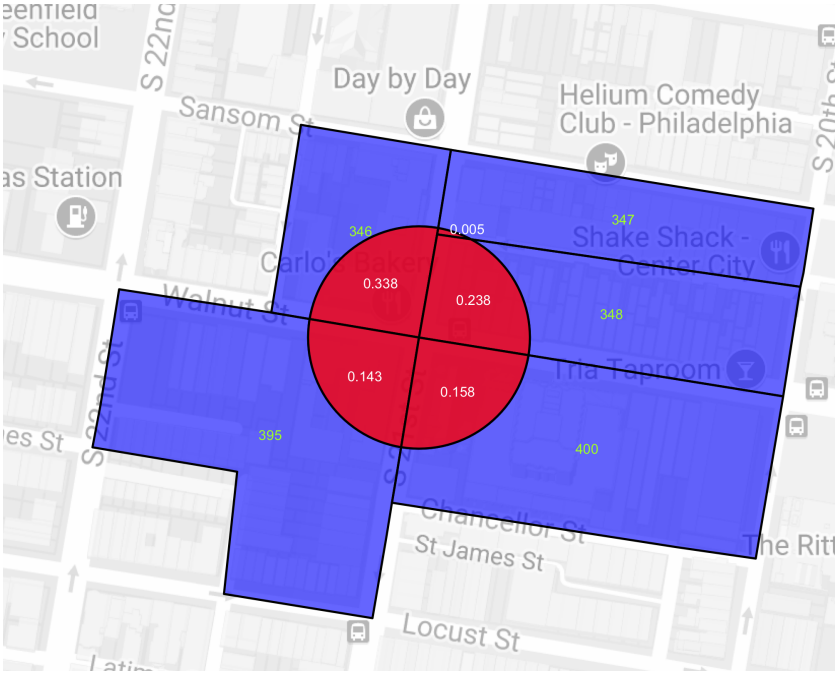


Figure 2.12: Our intersection at 21st and Walnut. The blue shapes are blocks, the red are the resulting polygons from intersecting each block with our intersection circle. The green text is the index of each block, and the white text is the area of the red segment divided by the blue, i.e. the proportion of the block inside our intersection circle.

of the surrounding blocks, using the estimated populations as our weights.

Business Data

In typical fashion, for each intersection circle we record which businesses are contained within, if any. We continue to use our ten categories of businesses detailed in section 3.2. We have a total of 72,020 businesses, with opening hours for 19,140 of them.

Crime Data

We treat our crime data similarly to the core analysis. We categorize murder, rape[7], assault and robbery as violent crime[8]; theft, burglary, arson, vandalism and disorderly conduct form

[7] The definition of Uniform Crime Reporting changed with respect to rape in 2013, changed by Robert Mueller. The new category is more general. For that reason, we combine "Sex Offences (not commercialized)" with rape to form one category.

[8] UCR codes 100, 200, 300, 400, 800 and 1700

'non-violent crime[9]. For our purposes, we exclude other crime categories.

The main analysis counts the number of the above two types that occur within 50 meters of each intersection from 2009 till July 2016, while the secondary sensitivity analysis counts from July 2016 till the start of 2018. This reflects the collection of the business data, which occured in July 2016. For each type, we also add the two to form the totals for each intersection. Note that these time segments are not the same as in the core analyses.

Landuse Data

We don't tend to use landuse data for our intersection level matching, because zoning is correlated with businesses, thus we're wary of matching on what would potentially be our treatment or outcome - this match could either be matching on an instrument, potentially inflating unmeasured biases, or could be a result of the treatment, i.e. zoning changing through application. This is not a problem for our first matching analyses, because when zoning is used as our treatment variable, businesses are not.

For the calcuation: similar to the demographic analysis, we can calculate the intersection of each landuse polygon with our circles, and e.g. measure the areas thus contained in our circles of each landuse type, or even the number of lots of each type.

[9]UCR codes 500, 600, 700, 900, 1400 and 2400

Chapter 3

Core Analysis of Urban Vibrancy in Philadelphia

3.1 Exploring Neighborhood Factors Associated with Safety in Philadelphia

3.1.1 Association between Crime and Population

We first examine whether population Count_i and/or Density_i are associated with either violent or non-violent crimes in Philadelphia. Figure 3.1 plots the relationship between these two population measures and violent vs. non-violent crimes. Figure 3.1 also include the correlation and test statistic for the slope from a robust regression that downweights outlying values (Huber 2011). We also explored Poisson and Negative Binomial regressions but found that these alternative formulations did not give substantially different results.

We see in Figure 3.1 that population count is more strongly associated with both violent crime and non-violent crime than population density. In fact, population density is not significantly associated with violent crime, and negatively associated with non-violent crime.

The lack of a strong positive association between population density and crime is especially notable in the context of popular historical hypotheses such as Simmel (2011) which argue that high population density leads to negative attitudes and hostility. In contrast, we find population size to be more strongly associated with crime compared to population density, which supports the work of Verbrugge and Taylor (1980).

In order to incorporate the association between crime and population count into our subsequent analyses, we define *excess violent crime* in each block group as the residual violent crime for that block group from the robust regression of violent crime on population count. Similarly, we define *excess non-violent crime* in each block group as the residual non-violent

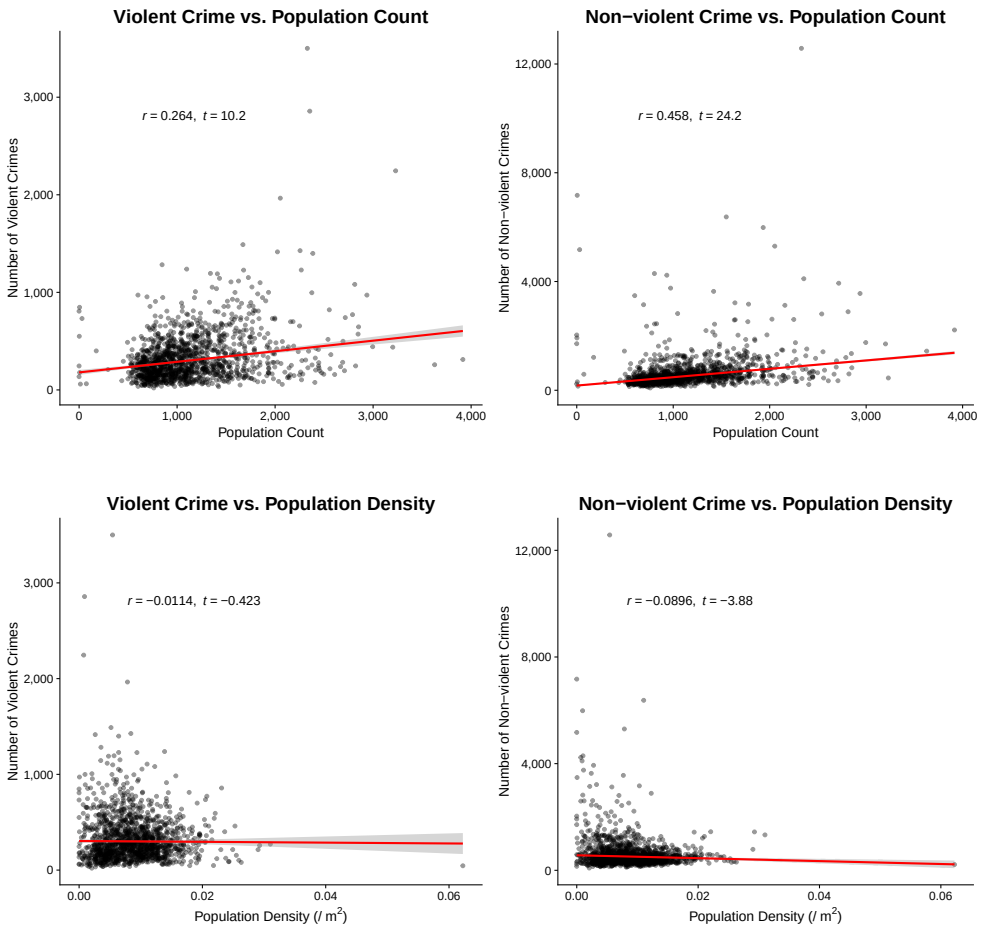


Figure 3.1: Association between Safety and Population. Predictor variables are either population count (top row) or population density (bottom row). Outcome variables are either violent crime counts (left column) or non-violent crime counts (right column). Red lines (and grey bands) are the least squares lines (and confidence bands) from a robust regression that downweights outlying values.

crime for that block group from the robust regression of non-violent crime on population count. In other words, the variables for safety in the next section 3.1.2 will be excess crime (violent or non-violent) beyond the expected crime based on population count.

3.1.2 Association between Excess Crime and Economic Measures

As outlined in section 2.2.2, we focus on two measures of the economic health of each block group in Philadelphia: 1. per-capita income and 2. our poverty metric. Figure 3.2 plots the relationship between these two economic measures and excess violent versus non-violent

crime.

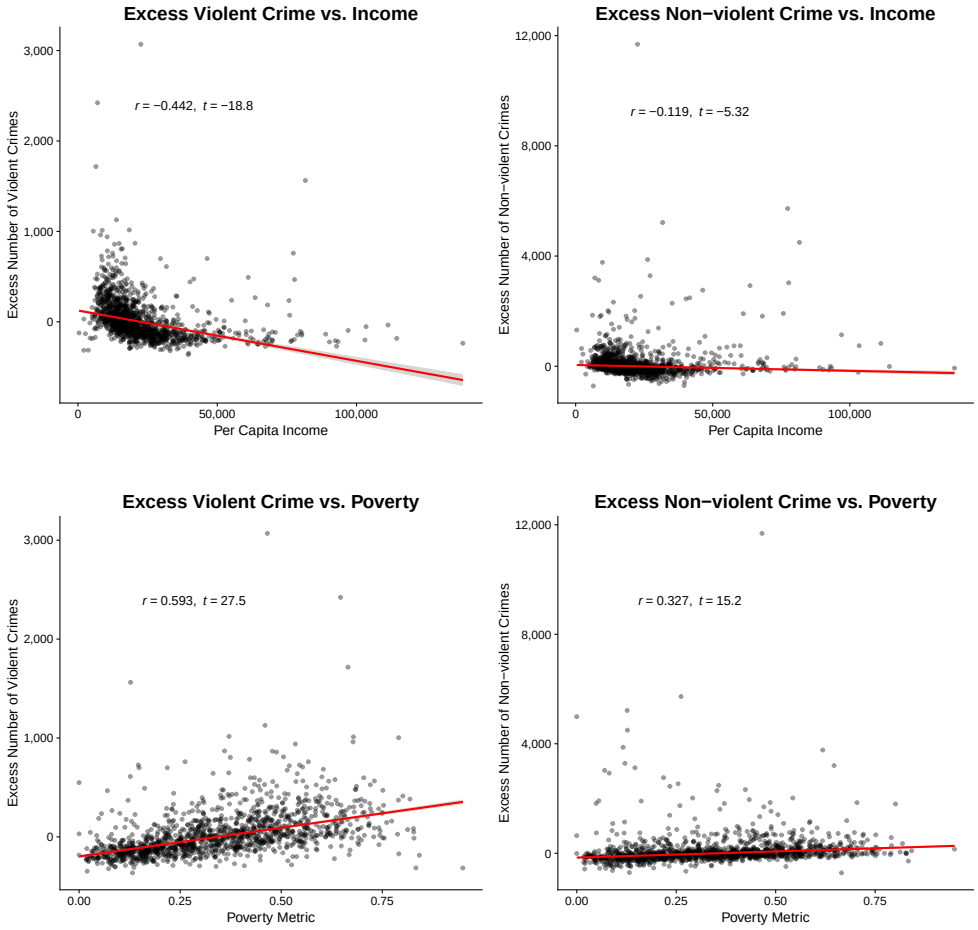


Figure 3.2: Association between safety and economic measures. Predictor variables are either per-capita income (top row) or our poverty metric (bottom row). Outcome variables are either excess violent crime counts (left column) or excess non-violent crime counts (right column). Red lines (and grey bands) are the least squares lines (and confidence bands) from a robust regression that downweights outlying values.

In Figure 3.2, we see a strong negative relationship between excess violent crime and income ($r = -0.44$) and a strong positive relationship between excess violent crime and poverty ($r = 0.59$). There is also noticeable non-linearity in the relationship between income and violent crime, with an even stronger linear relationship between violent crime and income for per-capita income below \$50,000 but much less of a relationship above per-capita income of \$50,000.

These economic measures have a much weaker relationship with excess non-violent crime. There is a weak negative association between per-capita income and excess non-violent crime ($r = -0.12$) and a weak positive association between poverty and excess non-violent crime ($r = 0.33$). Together these results suggest that per-capita income and poverty are strongly associated with excess violent crime but not excess non-violent crime, possibly because non-violent crimes are more crimes of opportunity occurring in areas located away from where the perpetrators of those crimes reside. Crimes of opportunity may be more driven by locations of businesses (rather than residences) which helps to motivate our work in sections 3.2-3.3.

In order to incorporate the association between crime and these economic measures into our subsequent analysis, we now re-define *excess violent crime* in each block group as the residual violent crime in that block group from the robust regression of violent crime on population count, per-capita income and our poverty metric; Similarly for *excess non-violent crime*. So for the next section 3.1.3, excess crime (violent or non-violent) in a block group is the number of crimes beyond expectation based on population count, income and poverty.

3.1.3 Association between Excess Crime and Land Use Zoning

Up to this point in our exploratory data analyses, we have focussed on the relationship between safety and features based on residents, namely the population and economic health, of each neighborhood. However, our primary goal is to investigate the role that the *built environment* of the neighborhood plays in safety, since effects of the built environment could inform future public policy initiatives.

As presented in section 2.2.3, one type of data that we have pertaining to the built environment is the land use zoning designations for each lot in the city of Philadelphia. We used those zoning designations to create three measures of the "vibrancy" in each block group i: the fraction of vacant land (Vacant.Prop_i), the fraction of mixed use land (MixedUse.Prop_i) and the ratio of commercial area to residential area (ComRes.Prop_i). Figure 3.3 plots the

relationship between these three land use vibrancy measures and excess violent versus excess non-violent crime.

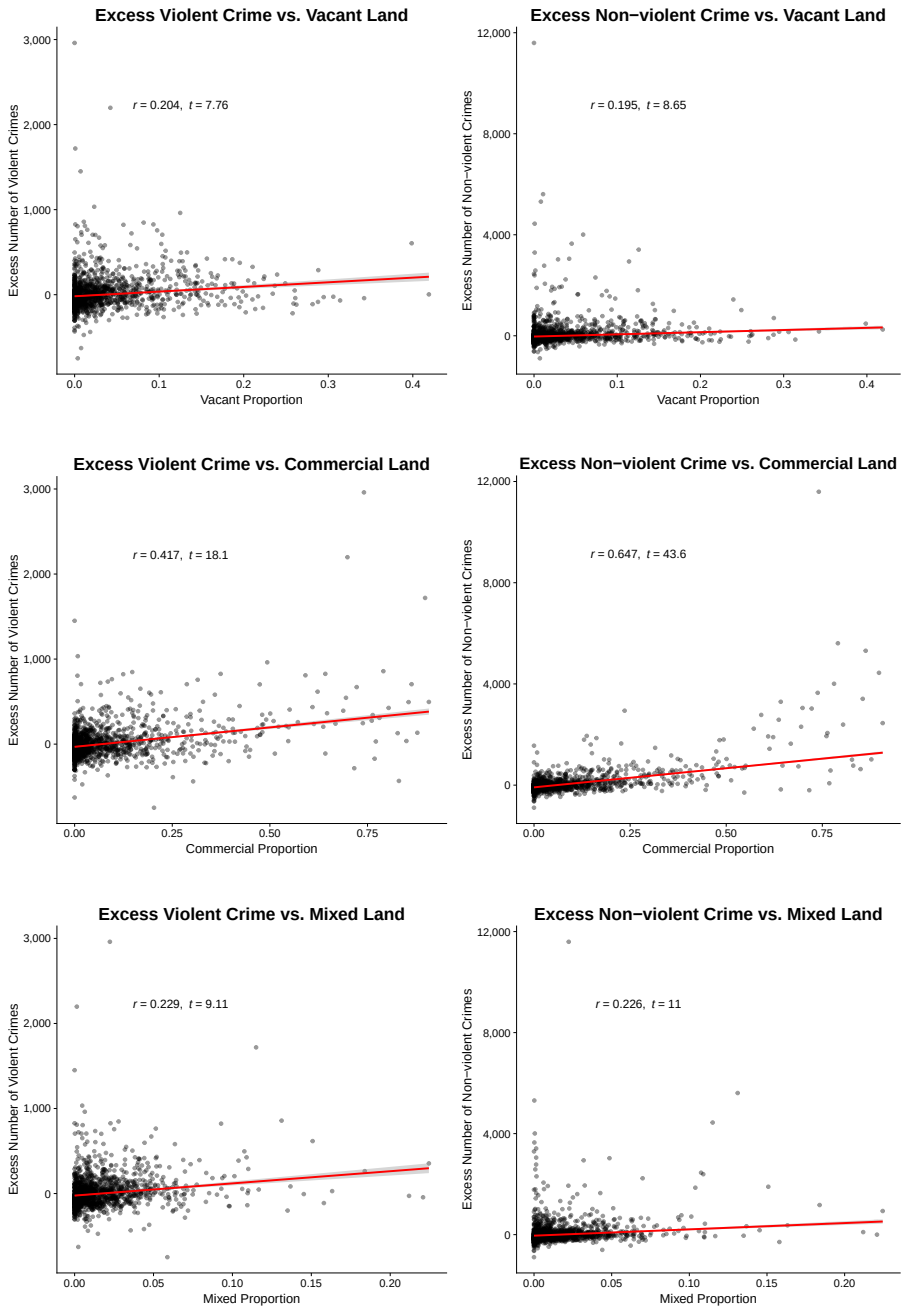


Figure 3.3: Association between safety and land use vibrancy measures. Red lines (and grey bands) are the least squares lines (and confidence bands) from a robust regression that downweights outlying values.

Examining Figure 3.3, we see a moderately strong positive relationship between vacant

proportion and excess violent crime ($r = 0.2$) and a similar relationship between vacant proportion and excess non-violent crime ($r = 0.20$). We see a stronger positive relationship between commercial vs. residential proportion and excess violent crime ($r = 0.42$) and an even stronger positive relationship between commercial vs. residential proportion and excess non-violent crime ($r = 0.65$). Finally, we see moderately strong positive relationship between mixed use proportion and excess violent crime ($r = 0.23$) and between mixed use proportion and non-violent crime ($r = 0.23$).

The moderately strong positive relationship we find between vacant lots and safety is related to recent investigations into the effect of "greening" of vacant lots on neighborhood safety (Branas et al. 2011). In that study, vacant lots that were randomly selected to be turned into green spaces were compared with a control set of vacant lots without an intervention. Branas et al. (2011) found that the "greening" of vacant lots was associated with a reduction of certain crime types and promotion of some positive health outcomes.

The strong positive relationship we find in Figure 3.3 between commercial proportion and crime is also very interesting in the context of contemporary theories of urbanism. As we describe in section 2.1, the "eyes on the street" theory of Jacobs (1961) and "natural surveillance" theory of (Deutsch 2016) argue that safer and more vibrant neighborhoods were those that have greater presence of people on the street achieved through a mixing of commercial and residential properties.

Our findings in Figure 3.3 do not support the idea that a mix of commercial and residential land use leads to increased safety. However, we must concede that land use zoning designations are a rather imperfect and low resolution indication of urban vibrancy. Land use zoning demonstrates a type of use deemed appropriate based on proximity to residents as well as governing fire safety and exposure. Zoning also regulates building heights and the type of allowable activity in a given structure. In particular, the zoning designation of a particular lot as commercial does not provide insight into whether the commercial enterprise located in that lot contributes positively to vibrancy of the area.

This type of data also does not contain information about whether that commercial enterprise is open or closed during times when crimes tend to be committed. As we describe in section 2.1, an indication of the activity of a commercial business is important to evaluating its impact on the neighborhood. An open business can encourage vibrancy through the activity of its staff and customers, or can give a sense of vacancy and isolation to an area if it is closed.

Thus, key information for urban vibrancy is missing from the land use zoning data, such as the types of business occupied on commercial property and when those businesses are open. This missing information motivates our investigation of more detailed measures of neighborhood vibrancy based on business data in the following section 3.2.

3.2 Urban Vibrancy Measures based on Business Data

As discussed in section 3.1.3, measures based on land use zoning designations are an insufficient summary of the vibrancy of a neighborhood. We can not evaluate whether a mix of commercial and residential properties promotes safety without first establishing what types of business enterprises are present in lots zoned for commercial use. We need to better understand when businesses are active, what type of business they are, and how they contribute to vibrancy. To that end, we outline our manual assembly and curation of a database of Philadelphia businesses, as well as the construction of several measures of business vibrancy from that data.

3.2.1 Business Data

We have manually assembled a database of Philadelphia businesses by scraping three different web sources: Google Places, Yelp, and Foursquare. Each of these sources provide the GPS locations for a large number of businesses in Philadelphia, as well as opening hours for a subset of those businesses.

The most difficult issues with assembling this business database are: 1. integrating these

three data sources and removing overlapping businesses and 2. categorizing all businesses into a small set of *business types*. Table 3.1 gives the number of businesses and the number of those businesses where we have opening hour information. We also give counts of the total number of businesses and the number of businesses with opening hours in the union of all three data sources (removing duplicates between data sources).

Table 3.1: Number of businesses and number with opening hours from each data source.

	Google	Yelp	Foursquare	Union
Total businesses	34,768	12,534	40,331	72,020
Businesses with hours	12,346	7,728	7,022	19,140

Each data source has its own categorization for businesses, with Google using about a hundred categories and Yelp and Foursquare each using closer to a thousand categories. Out of the myriad of business categories across all data sources, we created ten *business types*: Cafe (4,166), Convenience (1,481), Gym (1,273), Institution (24,489), Liquor (316), Lodging (461), Nightlife (5,108), Pharmacy (799), Restaurant (7,909), and Retail (31,147). The values in parentheses are the total number of businesses in each business type.

A particular business can belong to multiple business types, e.g. a restaurant that also sells liquor to go. Most of these business types are self-explanatory, but we need to clarify a few details. The cafe type includes cafes, bakeries and coffee shops that are not full restaurants. The restaurant type also includes meal delivery and meal take out businesses. Institution is a broad type that includes banks, post offices, churches, museums, schools, police and fire departments, as well as many others.

3.2.2 Measures of Business Vibrancy

We use our assembled business data to create several high resolution measures of business vibrancy at any particular location in the city of Philadelphia. We want these measures to encapsulate whether a given location has a concentration of a businesses of a particular type, and whether those businesses are active storefronts (i.e. open) during times of the week when crimes tend to be highest.

We examined the frequency of crimes at different times of the week and isolated two "high crime windows" that have a disproportionately large number of crimes (both violent and non-violent) relative to other times of the week. These two high crime windows are *weekday evenings*, which we define as 6-12pm Monday-Friday and *weekend nights*, which we define as 12-4am Saturday-Sunday.

In section 3.3, we will evaluate whether business vibrancy is associated with crime totals during these two specific high crime windows as well as throughout the entire week.

The first set of measures of business vibrancy we consider are simply the total number of businesses of each *business type* near to any particular location in the city. Recall that our ten business types are Cafe, Convenience, Gym, Institution, Liquor, Lodging, Nightlife, Pharmacy, Restaurant, and Retail. We expect that some of these business types will be more associated with safety than others.

In addition to the total number of businesses of each type near to a particular location, we want to take into account whether those businesses are active storefronts in the sense of being open. In particular, we are interested in whether a given location has businesses of a particular type (e.g. cafes) that are open longer than expected.

We first establish a *consensus* number of open hours for each business type by calculating the average open hours across all businesses of that type in Philadelphia. We can then calculate the *excess* number of open hours for each business (for which we have open hours) in Philadelphia relative to the consensus for that business type. For example, a particular cafe will have *excess* of 2 if it is open for 2 hours more than the consensus open hours for all cafes in Philadelphia, whereas a particular restaurant will have *excess* of -3 if it is open for 3 hours less than the consensus open hours for all restaurants in Philadelphia.

Building upon these calculations, the second set of measures of business vibrancy we consider are the average excess hours of businesses of each *business type* near to any particular location in the city. We calculate these excess hour measures over the entire week as well

as just within in the two high crime windows (weekday evenings and weekend nights).

In summary, we have two sets of measures of the business vibrancy around any particular location: the number of businesses of each business type and the average excess hours of each business type. The latter can be calculated over the entire week or just within the high crime windows mentioned above.

In section 3.3, we evaluate the association between these business vibrancy measures and both excess violent and non-violent crimes within the local neighborhoods defined by our census block groups.

3.3 Evaluating the Association between Business Vibrancy and Safety

With our new business vibrancy measures in hand, the goal of our analysis is evaluating the association between these measures and safety at the local neighborhood level, while controlling for the characteristics of those neighborhoods.

We will control for neighborhood characteristics by focusing our analyses on comparing pairs of locations within each block or within each block group. Underlying this strategy is an assumption that the census blocks or block groups are small enough that different locations within these areal units (blocks or block groups) should be highly similar with regards to the demographic and economic measures we examined in Sections 3.1.1 and 3.1.2.

We explore two types of within-block comparisons. In section 3.3.1, we find pairs of locations within block groups where one location has businesses that are "open longer" relative to the consensus for their business type and where the other location has businesses that are "open shorter" relative to the consensus for their business type. We then examine these within-block-group pairs to see if there are differences in crime between "open shorter" vs. "open longer" locations.

In section 3.3.2, we find pairs of locations within blocks where one location has the highest

density of crimes and the other location has the lowest density of crimes within that block. We then examine these within-block pairs to see if there are differences our business vibrancy measures between the "high crime" vs. "low crime" locations.

3.3.1 Comparing "Open Shorter" vs. "Open Longer" Locations

For each of our ten business types, we identify block groups that contain a pair of businesses (of that type) where one of those businesses has long opening hours and the other business has short opening hours. We define a business as having long opening hours if its total opening hours are above the 75th percentile for businesses of that type. Similarly, we define a business as having short opening hours if its total opening hours are below the 25th percentile for businesses of that type.

We further restrict ourselves to block groups where the pair of businesses are at least 140 meters apart, which is roughly the size of a Philadelphia city block. It should be noted that only a small subset of the 1,336 block groups in Philadelphia will contain such a valid pair of businesses: both a long opening and short opening business of a particular type separated by at least 140 meters. As an example, lodging had only one block group in the entire city with a valid within-block group comparison for the total week comparison and so this business type is excluded from this analysis.

For each block group containing such a valid comparison, we then count the number of crimes that occurred within a 70 meter radius around both the long opening hour business and the short opening hour business (which ensures that we do not double count any crimes for both businesses).

The object of this analysis is the difference in crimes between the short opening hour business and the long opening hour business within each block group that contains such a business pair. If businesses that are active (open for a longer period) help to deter crime and promote safety, then these differences in crime should be positive. For each business type, we calculate a matched pairs mean differences in crime around the short opening hour

minus long opening hour businesses in each within-block group pair.

In Figure 3.4, we display the matched pair mean differences in crime between short opening and long opening hour businesses of each business type separately. We calculate different matched pair mean differences for only violent crimes, only non-violent crimes and all crimes. The significance threshold for these t-statistics was Bonferroni-adjusted to account for the number of comparisons being tested. We also divide up these comparisons into the three different time windows discussed in section 3.2.2: the entire week plus two high crime windows, *weekday evenings* and *weekend nights*.

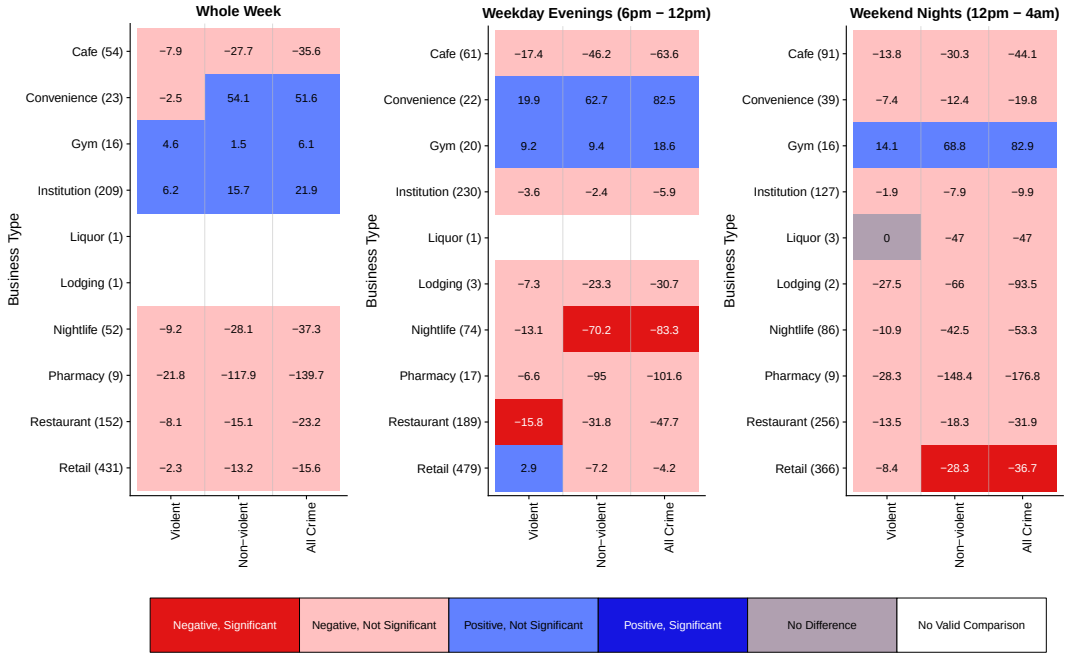


Figure 3.4: Matched pair mean differences in crime between short opening and long opening hour businesses, calculated separately for each combination of crime type and business type. Different panels are used to display the mean differences calculated over the entire week vs. just weekday evenings vs. just weekend nights. The significance threshold of $p = 0.05$ was Bonferroni-adjusted to account for multiple comparisons. Values in parentheses are the number of block groups with valid comparisons for that business type

Examining Figure 3.4, we see mostly negative differences (red) which imply that more crimes are occurring around the business location with longer open hours, especially nightlife locations and restaurants. A notable exception are gyms, which show positive differences which imply fewer crimes occurring around the gym with longer open hours, for all windows

and both crime types.

Violent crimes near convenience locations are also an interesting case. Over the entire week, fewer non-violent crimes occurred around the convenience locations with longer open hours (blue), while approximately the same number of violent crimes occurred. For the weekday evening window, far fewer crimes of both types occurred near convenience locations with longer open hours (blue), but this trend is reversed during weekend nights, where more crimes occurred near convenience locations with longer open hours.

Recall that our preliminary hypothesis, motivated by Jacobs (1961) and Deutsch (2016), was that greater business vibrancy would be associated with fewer crimes around those vibrant locations relative to less vibrant locations in the same block group. The results in Figure 3.4 for gyms does show a trend in this expected direction, but the results for most other business types goes against that hypothesis.

That said, there are not many differences in Figure 3.4 that are statistically significant. To a large extent, the lack of significance is driven by the small sample sizes in these comparisons. For example, there are only nine block groups with a pair of open shorter vs. open longer pharmacies for the whole week comparison, which does not give us much power to detect differences in crime associated with differences in business vibrancy.

Another weakness of this analysis is that we picked our locations for these comparisons based on a single "longer" open business and a single "shorter" open business. To incorporate a greater number of businesses in our comparisons, we can instead focus on comparing locations based on high versus low crime in the next section 3.3.2.

3.3.2 Comparing "High Crime" vs. "Low Crime" Locations

In this comparison, we first calculate the location with the highest crime frequency and the location with the lowest crime frequency within each block. We perform this analysis on the census block level (rather than the block group level) in order to give an even higher resolution view of the association between vibrancy and safety. For each business type

separately, we then calculate our measures of business vibrancy from section 3.2.2 around both the high crime and low crime locations within each block.

Many blocks do not contain any businesses of particular business types near either high or low crime locations, which excludes those blocks from any comparisons involving that particular business type. We further restrict ourselves to blocks where the highest crime and lowest crime locations are at least 100 meters apart. Similar to section 3.3.1, these restrictions limit the sample size for each of our comparisons.

For each block containing such a valid comparison, we calculate our two measures of business vibrancy, the number of businesses of each business type and the average excess hours of each business type, around the high crime and low crime locations in those blocks. For each business type, we calculate a matched pairs t-statistic for differences in the business vibrancy measures around the low crime location minus the the business vibrancy measures around high crime location within each block. If business vibrancy helps to deter crime and promote safety, then these differences in business vibrancy should be positive.

In Figure 3.5, we display the matched mean differences in the two business vibrancy measures (the number of businesses of each business type and the average excess hours of each business type) between the low crime and high crime within-block locations. We calculate differences for locations based on violent crimes and locations based on non-violent crimes. The significance threshold for these t-statistics was Bonferroni-adjusted to account for the number of comparisons being performed. We again also divide up these comparisons into the three difference time windows discussed in section 3.2.2: the entire week plus two high crime windows, *weekday evenings* and *weekend nights*.

We see in Figure 3.5 that the number of businesses difference is significantly negative (red) for both violent and non-violent crimes for essentially all business types, most strongly retail stores and restaurants. This result suggests that there are more businesses around the higher crime locations than the lower crime locations.

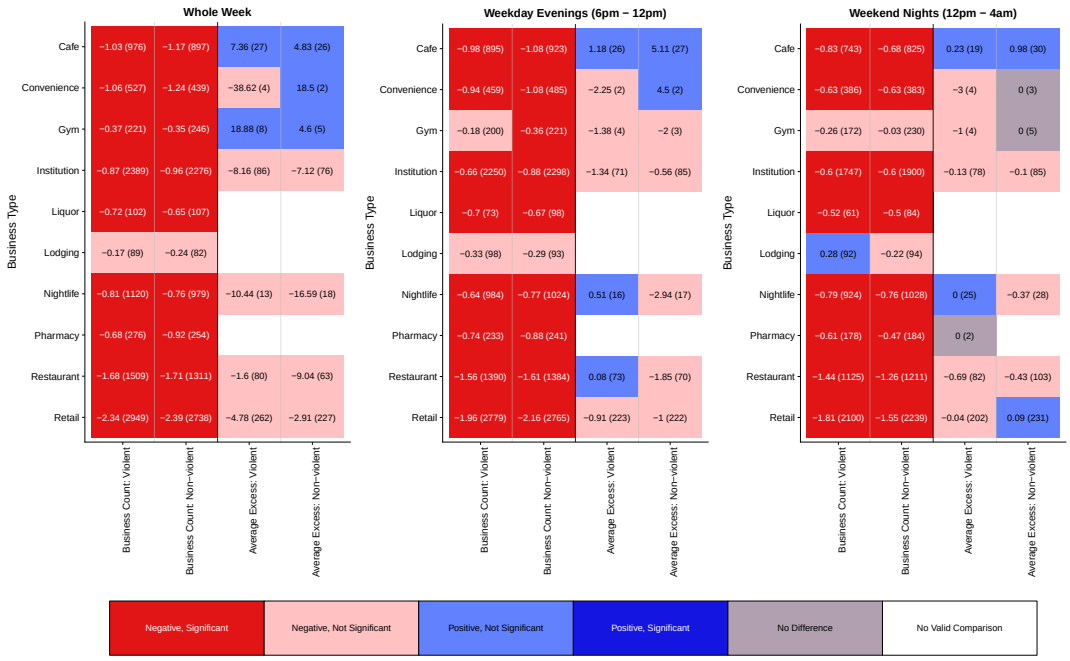


Figure 3.5: Matched pair mean differences in measures of business vibrancy between high crime and low crime locations, calculated separately for each combination of crime type and business type. Different panels are used to display the mean differences calculated over the entire week vs. just weekday evenings vs. just weekend nights. The significance threshold of $p = 0.05$ was Bonferroni-adjusted to account for multiple comparisons. Values in parentheses are the number of block with valid comparisons for that business type.

However, we also observe in Figure 3.5 that for many of these business types, there are positive differences (blue) for our average excess hours metric, which implies that those businesses are open longer around the low crime location compared to the high crime location. These differences are not as significant, but we still see evidence of an interesting and subtle finding: more crimes tend to occur near business locations but fewer crimes tend to occur near businesses that are open longer, for cafes, and gyms. Note that the left hand plot, Whole Week, contains the largest comparison in terms of crimes and hours counted.

We can also compare our original land use zoning measures of vibrancy from section 3.1.3 between these high and low crime locations. We again calculate differences for locations based on violent crimes and locations based on non-violent crimes, but now the differences are based on our three land use vibrancy measures: the fraction of vacant land, the fraction of mixed use land and the ratio of commercial area to residential area.

In Figure 3.6, we display the matched mean differences in the three land use vibrancy measures between the low crime and high crime within-block group locations. We again also divide up these comparisons into the three time windows discussed in section 3.2.2: the entire week plus two high crime windows, *weekday evenings* and *weekend nights*.

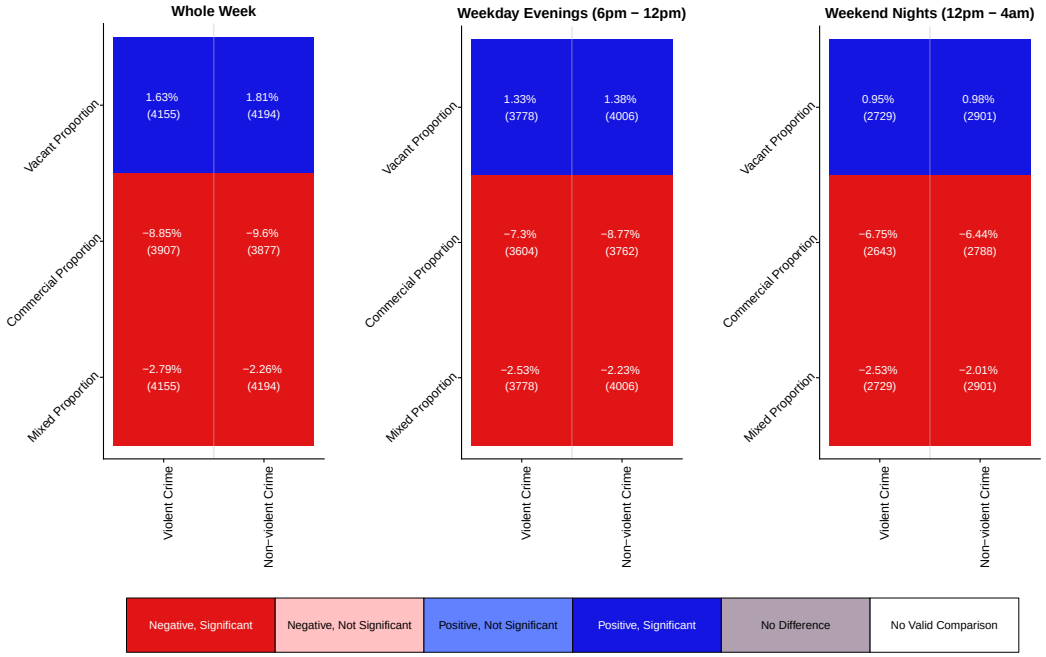


Figure 3.6: Matched pair mean differences in measures of land use zoning vibrancy, the fraction of vacant land and the ratio of commercial area to residential area, between high crime and low crime locations (calculated separately for each crime type and business type). Different panels are used to display the mean differences calculated over the entire week vs. just weekday evenings vs. just weekend nights. The significance threshold of $p = 0.05$ was Bonferroni-adjusted to account for multiple comparisons. Values in parentheses are the number of block groups with valid comparisons for that business type.

In Figure 3.6, we see very strong negative differences for mixed proportion and commercial vs. residential proportion, both of which strongly suggests that there is more mixed zoning and zoning for commercial use near to the high crime locations. This association between commercial enterprise and safety was also observed in section 3.1.3 and motivated our development of more detailed measures of business vibrancy in section 3.2.

We also see very strong positive differences for the vacant land proportion which suggests the presence of more vacant land near to low crime locations compared to the high crime

locations. This finding is notable when compared to the positive association between vacant proportion and crime seen in Figure 3.3.

Together, those two findings suggest that neighborhoods with more vacant properties overall have higher crime but when looking within those neighborhoods, crimes tend not to be located near vacant properties. These results are especially interesting given the mixed effects on crime resulting from the "greening" of vacant lots in the study by Branas et al. (2011).

3.3.3 Summary of Business Vibrancy and Safety

Our analysis pipeline for studying the association between business vibrancy and safety has produced several findings that could impact current evaluations of contemporary theories in urban planning. First, we find that more crimes occur near business locations but that businesses that are open for longer periods are associated with fewer crimes. Second, we find that although neighborhoods with more aggregate vacancy have higher crime (section 3.1.3), when comparing locations within each neighborhood, crimes tend not to be located near vacant properties.

Another important observation from Figures 3.4 and 3.5 is the substantial heterogeneity in the association between business vibrancy and crime both across different business types and different time windows. The power of both studies was compromised by small sample sizes as there are only a limited number of block groups that permit a pair of comparable locations. The associations between land use zoning and safety in Figure 3.6 are more significant due to much larger sample sizes of locations for these comparisons.

Clearly, the associations between safety and neighborhood vibrancy are subtle, heterogeneous, and in need of even higher resolution studies to fully understand. In section 3.4, we discuss alternative strategies for matched comparisons that may permit more high resolution (and large sample size) analyses.

3.4 Discussion of Core Urban Vibrancy Analysis

The recent availability of high resolution data on cities provides a tremendous opportunity for sophisticated quantitative evaluation of historical and current urban development. To aid in this effort, we outline a framework for data collection and analysis of the associations between safety, economic and demographic conditions and the built environment within local neighborhoods. We used this framework to investigate a more specific goal: the creation of quantitative measures of "vibrancy" based on the built environment of a neighborhood and exploration of the association between these vibrancy measures and neighborhood safety.

We find that population density is not strongly associated with either violent or non-violent crime, which argues against the theory of Simmel (2011). We find that population count is a more important predictor of crime, which supports the work of Verbrugge and Taylor (1980). We also explored the association between crime and economic measures as well as measures of vibrancy derived from land use zoning data, but found that these measures were not an adequate summary of the local commercial vibrancy of an area.

To address vibrancy at a higher resolution, we constructed several measures of business vibrancy and employed matching of locations within block groups to evaluate the relationship between business vibrancy and safety. Our business vibrancy measures (number of businesses and average excess hours of businesses) are designed to be proxies for the "eyes on the street" concept of Jacobs (1961).

Our results suggest that more crimes occur near business locations but that businesses of some types that are active (open) for longer periods could be associated with fewer crimes. We also found that the overall proportion of vacancy in a neighborhood is associated with higher crime but that within a neighborhood, crimes tend to not occur near to vacant properties.

We also found substantial heterogeneity in the direction and strength of the association between crime and business vibrancy across different business types and different times of

the week. Our view of the business vibrancy in a local area could be possibly improved by incorporating additional information such as more direct measures of business activity (beyond being open or not) when that data is available. Another potential option is business ratings, which are a primary feature of one of our commercial data sources, Yelp.

It may also be possible to perform our matching analyses at a higher resolution level, such as individual streets, rather than just locations within the same block group which may have more power for detecting subtle relationships. For example, Weisburd (2015) focussed on the street segment as their geographical unit of analysis when studying the concentration of crime.

It should also be noted that our simple testing procedures in section 3.3 do assume that crimes are realized independently. This assumption is tenuous when there are multiple crimes reported from the same incident or dependence within perpetrators for repeated crimes and between co-perpetrators. However, we do not believe that these dependencies have had a substantial effect on our comparisons.

Outside of the business vibrancy measures that are the focus of this paper, there are many alternative data sources that would help to further define the vibrancy of local urban areas. Home and property prices are a valuable resource for modeling the desirability of a neighborhood; we add such data in our augmented analyses in chapter 5.

The company *Walk Score* produces a composite measure of the walkability of a neighborhood but their measure does not include several important details (Goodyear 2012), such as the types of available businesses which we found to be relevant in section 3.3. The direct measure of foot traffic at the neighborhood or street level would certainly improve measures of urban vibrancy but this data is also not currently publicly available.

We encourage the adaptation of our analysis pipeline to other research questions within the urban analytics community. The code and public data that were used in our analyses is available as a github repository at: `https://github.com/ColmanHumphrey/urbananalytics`

Chapter 4

Matching: Generation, Evaluation and Selection

In this section, we detail our Full Selection Procedure: how to start with a set of units and end with a single best match.

We first define matches and matching distances in section 4.1, along with various options for defining distance in section 4.2.

Sections 4.3 and 4.4 detail how to evaluate and select matches, while section 4.5 relates our procedures back to the goals of causal inference. Section 4.6 gives our method to generate a set of matches for a given observational setup, in order to create a set of matches to select.

Section 4.7 extends the generation and selection procedures to allow flexibility in deciding the size of the matched set. This relates back to the feasible average treatment effect on the treatment from section 1.1.2 in the introduction. Section 4.8 gives details on how to extend our methods to non-bipartite matches.

We test our procedure in a simulation study in section 4.9, and wrap up this chapter with a discussion section, section 4.10.

4.1 Minimising Distances to Form Matches

4.1.1 Similarity and Distance

In matching, we want to do analysis with units that are "similar", or equivalently with units that minimise distance.

We have to decide what we mean by similar. We use two main concepts of similarity: purely covariate based similarity, and similarity as it relates to the treatment.

The most common form of judging covariate similarity is to use Mahalanobis distance. This

is a generalised version of measuring distance between two units in standard deviations.

The most common form of treatment similarity is to match on the propensity score - the probability of receiving the treatment given the covariates. We want matched units to be close on this metric.

Matching Matrix

There are many ways to form matched groups. In what follows, the majority of our methods are adaptable to most grouping structures, although we will generally assume we're forming matched pairs.

A matched pair is a pair of indices (i, j) such that $T_i + T_j = 1$ (exactly one is treated). Let $\boldsymbol{M}$ refer to a set of matches represented as a matrix with two columns, so that each row is a pair. Within each row, we assume the first column is the treated indices, and the second are the matches control incides. Letting $\boldsymbol{M}_k$ refer to the kth row of $\boldsymbol{M}$, this means $T_{\boldsymbol{M}_{k,1}} = 1$ and $T_{\boldsymbol{M}_{k,2}} = 0$.

With matched pairs, we only allow each treated unit to appear at most once, thus the elements of the first column of $\boldsymbol{M}$ are unique. We don't always require the same of the control column.

In order to uniquely define the match matrix, we'll generally assume the first column is ordered, i.e. $\boldsymbol{M}_{k,1} < \boldsymbol{M}_{l,1}$ for all $k < l$.

For an example, say we have ten units to be matched, with units $2, 3, 6$ and 9 being treated. We decide the best controls are respectively $8, 4, 1$ and 4, i.e unit 4 is the control for both

unit 3 and unit 9. The match matrix $\boldsymbol{M}$ is thus:

$$\boldsymbol{M} = \begin{bmatrix} 2 & 8 \\ 3 & 4 \\ 6 & 1 \\ 9 & 4 \end{bmatrix} \tag{4.1}$$

Matching Distance

Once we have our distance metric[1], or rather distance premetric, we can define an $N \times N$ distance matrix D such that:

$$D_{i,j} = d(\mathbf{x}_i, \mathbf{x}_j)$$

For bipartite matches, we usually work with a subset of D in which we throw away control rows, i.e. delete row i for all i such that $T_i = 0$, and throw away treated columns, i.e. delete row j for all j with $T_j = 1$. The (k, l) element of this smaller matrix will be the distance between the kth treated unit and the lth control unit.

4.1.2 Forming Matches: Minimising Distance

Once we have a distance matrix D from a given distance premetric d, we can form our match: it's the set of pairs that minimises this pairwise distance.

Letting $\boldsymbol{M}$ refer to the resulting match matrix $\boldsymbol{M}$, for ease of notation we let $D_{\boldsymbol{M}_i} = D_{(\boldsymbol{M}_{i,1}, \boldsymbol{M}_{i,2})}$, the distance between the two units in the ith row.

The match is thus:

$$\underset{\boldsymbol{M}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N_t} D_{\boldsymbol{M}_i} : \boldsymbol{M}_{k,1} < \boldsymbol{M}_{l,1} \text{ for } k < l \right\} \tag{4.2}$$

[1]Not actually a metric. We certainly want $d(x, y) \geq 0$ and $d(x, y) = d(y, x)$, but we have a seminorm instead of a norm quality in the sense that $d(x, x) = 0$ for all x, while it's possible $d(x, y) = 0$ for some $x \neq y$. Further, we don't require the triangle inequality.

This defines a unique solution up to ties. Unless otherwise specified, we will select among ties randomly.

For the purposes of notation, define $D(\boldsymbol{M})$ as the total matching distance:

$$D(\boldsymbol{M}) := \sum_{i=1}^{N_t} D_{\boldsymbol{M}_i} \tag{4.3}$$

Note that distances are only identified up to a multiplicative constant. That is, multiplying all distances by the same constant will result in the same match. This is because the minimiser of the sums of distances is also the minimiser of that sum multiplied by any positive constant.

Matching With Replacement

With no extra conditions imposed in equation 4.2, the solution allows for controls to be used multiple times. This is often called Nearest Neighbour matching. Another simple way to define this matching solution is to find the control closest to each treated unit in terms of distance.

Finding the best match with replacement is fast, $\mathcal{O}(N_t N_c)$[2].

Matching with replacement is in general a bias-minimising solution, potentially at the expense of variance.

Optimal Matching

Optimal matching does not allow controls to be used multiple times. Its solution is thus:

$$\underset{\boldsymbol{M}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N_t} D_{\boldsymbol{M}_i} : \ \boldsymbol{M}_{k,1} < \boldsymbol{M}_{l,1} \text{ for } k < l, \ \boldsymbol{M}_{r,2} \neq \boldsymbol{M}_{s,2} \ \forall r \neq s \right\} \tag{4.4}$$

Finding the optimal match is much more costly than matching with replacement, it is $\mathcal{O}(N^3)$.

[2] Or just $\mathcal{O}(N^2)$ if that's more useful

Greedy Matching

Greedy matching also does not allow controls to be used multiple times.

Like optimal matching, it has equation 4.4 defining its target, but only tries to find an approximate solution to it: it searches through treated units one by one, adding the closest control unit at a given step, but then blocks that control unit from further matches.

There are many ways to search greedily in this fashion: search through the treated units randomly; add the control that creates the smallest pair at each step; if matching on a one-dimensional variable, search in some order of that variable.

Greedy search is also very fast, $\mathcal{O}(N_t N_c)$, but has the downside of being dominated as a solution by optimal matching. It should only be used when optimal matching is too expensive, but variance concerns rule out matching with replacement.

4.2 Defining Matching Distance

In this section, we define the distance functions we incorporate into our procedure.

4.2.1 Mahalanobis Matching

Let Σ be the covariance matrix of $\boldsymbol{X}$. The Mahalanobis distance is:

$$d_{\Sigma}(\mathbf{x}_i, \mathbf{x}_j) := (\mathbf{x}_i - \mathbf{x}_j)' \sum^{-1} (\mathbf{x}_i - \mathbf{x}_j) \tag{4.5}$$

Or commonly:

$$d_{\Sigma}(\mathbf{x}_i, \mathbf{x}_j) := \sqrt{(\mathbf{x}_i - \mathbf{x}_j)' \sum^{-1} (\mathbf{x}_i - \mathbf{x}_j)} \tag{4.6}$$

Let this define the $N \times N$ matrix D, i.e. $D_{i,j} = d(\mathbf{x}_i, \mathbf{x}_j)$. We allow the reader to decide if they want to take the square root or not.

The "Mahalanobis match" forms the set of pairs that minimises the total Mahalanobis distance.

In real data, we don't have the true value of Σ, the covariance matrix. We use $\hat{\Sigma}$, the estimated covariance matrix, in its place.

Rank-based Mahalanobis

As Rosenbaum (2002) points out, Mahalanobis distance is great for multivariate normal data, but can have undesirable properties for e.g. long-tailed data, or binary data, especially rare binary outcomes.

In its place, we use rank-based Mahalanobis distance. We convert each covariate to its ranks, with averages for ties, and we scale the resulting covariance matrix so that it has a constant diagonal. Thus we avoid the issue of outliers and long-tailed distributions. The second step is only relevant for variables with ties, and avoids the issue of e.g. rare binary variables creating huge differences with small variances.

If one desires, one could choose which variables to convert to ranks and which to keep in their raw form. For example, a bimodal continuous variable could be a candidate for leaving in its raw form[3].

Minor note on computation: fast methods of computing Mahalanobis distance matrices, such as Cholesky decompositions, perform better with all variables on a similar scale. This is done naturally if all variables are rank-converted. Finally, we divide all ranks by N, the total number of units, so that ranks are from zero[4] to one.

[3] Take for example a variable uniform over $[1, 2] \cup [4, 5]$. Numbers close to 2 and close to 4 will have close ranks but will be far apart; numbers close to 1 and numbers close to 2 won't be that far apart but will have far ranks.

[4] Or really $1/N$

Weighted Mahalanobis Matching

Both standard Mahalanobis matching and rank-based Mahalanobis matching weigh all variables equally, which may not be what we want. We may either have prior knowledge on variable importance, or we may wish to use a data-driven process to get the variable weights.

Adding weights to adjust Mahalanobis distance is straight forward. Let $\boldsymbol{w}$ be a vector of weights corresponding to the covariates, and $\boldsymbol{W} = \text{diag}(\boldsymbol{w})$ be the diagonal matrix containing these weights. Then the weighted Mahalanobis distance is given by:

$$d_{\boldsymbol{w},\Sigma}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)' \boldsymbol{W} \sum^{-1} \boldsymbol{W} (\mathbf{x}_i - \mathbf{x}_j) \tag{4.7}$$

As with standard Mahalanobis matching, in practice we use $\hat{\Sigma}$ in place of Σ, or the adjusted version from rank Mahalanobis. Further, readers may prefer to use $\sqrt{d_{\boldsymbol{w},\Sigma}}$.

4.2.2 Propensity Score Matching

Propensity score matching (Rosenbaum and Rubin 1983) first predicts the probability of treatment, and then uses these predictions as the matching variable.

The propensity score is defined as the probability of receiving the treatment given the covariates:

$$e(\mathbf{x}) = \mathbf{P}(T = 1 \mid \mathbf{x}) \tag{4.8}$$

This gives us our definition of propensity distance:

$$d_{\mathbf{e}}(\mathbf{x}_i, \mathbf{x}_j) = \Big| e(\mathbf{x}_i) - e(\mathbf{x}_j) \Big| \tag{4.9}$$

In similar fashion to the Mahalanobis cases, one could square this function if desired.

Of course in practice we don't know the true value of the propensity score, and must estimate

$\hat{e}(\mathbf{x})$ from the data.

Algorithm Choice

Typically propensity scores are estimated using logistic regression. This leads to a potential alternative propensity distance definition:

$$d_{\mathbf{e}}(\mathbf{x}_i, \mathbf{x}_j) = \Big|\text{logit}(e(\mathbf{x}_i)) - \text{logit}(e(\mathbf{x}_j))\Big| \tag{4.10}$$

Where $\text{logit}(p) = \frac{p}{1-p}$.

Lee, Lessler, and Stuart (2010) show a benefit to using more advanced "machine learning" methods such as random forests to estimate propensity scores. The benefit is significant when the true propensity model is not a linear additive function of the covariates. The cost of such methods (beyond computation time!) seems minimal when the logistic model is correct.

In our models, we generally use gradient boosting with shallow trees. Currently, Xgboost (Chen and Guestrin 2016) is an excellent implementation, available in many languages.

In-sample vs Out of Sample

An in-sample fit for propensity scores is when we use the data to train our model, and then plug the same data in for our predictions.

Some studies have shown that the estimated propensity score performs better than the true propensity score. This is usually ascribed to estimated scores "overfitting" the covariates, and thus producing better than random balance. We discuss this further in section 4.5.

This will happen with nearly all in-sample propensity score fits: the errors in the estimated scores will be correlated among units with similar covariates. For example, if the true fit is $e(X) \equiv 0.5$, a logistic fit will still produce some non-zero coefficients. Two units with similar covariates will likely have their probabilities estimated on the same side of 0.5, increasing

the chances of a match. The downside of this effect is that misspecified models have their biases magnified with in-sample fits.

Any methods that are given the ability to overfit, such as neural networks or forests, can have an even worse version of this effect. When using matching with replacement, models with large degrees of freedom can tend towards using a very small set of controls repeated many times when using in-sample fits.

These effect won't disappear but are reduced when out of sample propensity scores are used. We don't generally have true out of sample predictions, so we use some version of cross-validation to generate predictions[5]. The difference is minor for logistic regression, especially when N_t and N_c are large relative to the number of covariates.

When using caliper matching, the advantages of in-sample overfitting are less relevant.

4.2.3 Caliper Matching

Depending on your perspective, caliper matching is either propensity matching with a sprinkling of Mahalanobis matching, or vice versa.

Strict Caliper Constraint

The most traditional use of caliper matching is to only allow matches that are close in terms of estimated propensity score, i.e. within some specified caliper, and use Mahalanobis matching to match units once the propensity score difference satisfies the caliper constraint.

Given a caliper width δ, we can define this caliper distance:

$$d_{\delta,\infty}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} \infty & d_{\mathbf{e}}(\mathbf{x}_i, \mathbf{x}_j) > \delta \\ d_{\Sigma}(\mathbf{x}_i, \mathbf{x}_j) & d_{\mathbf{e}}(\mathbf{x}_i, \mathbf{x}_j) \leq \delta \end{cases} \tag{4.11}$$

[5]The most straight-forward is $k-$fold cross validation. We would split our data into e.g. five equal sized groups, usually stratifying by the treatment variable. For each quintile, we train a model on the other four and use that for prediction. Thus no unit's score is predicted from a model that used it for training.

Setting $\delta = 1$[6] results in no penalisation, and therefore gives exactly Mahalanobis distance.

Soft Caliper Constraint

There are two issues with strict constraints: for a given caliper width δ, you may not be able to form a full set of matches[7]; within the specified caliper, propensity scores are ignored, instead of further informing the match.

We can solve these problems by introducing a ramp function penalisation. That is, we multiply the propensity violation by some number, and add that to the Mahalanobis distances.

Given a caliper width δ and a multiplicative factor λ, we can define this caliper distance:

$$d_{\delta,\lambda}(\mathbf{x}_i, \mathbf{x}_j) = d_\Sigma(\mathbf{x}_i, \mathbf{x}_j) + \lambda \max\{d_\mathbf{e}(\mathbf{x}_i, \mathbf{x}_j) - \delta, 0\} \tag{4.12}$$

Setting $\lambda = \infty$ recovers the strict constraint.

Like the strict constraint, setting δ large enough recovers Mahalanobis distance. Further, setting $\lambda = 0$ also recovers Mahalanobis distance.

Setting $\delta = 0$ makes the distance a linear combination of Mahalanobis matching and propensity score matching. As referenced in section 4.1.2, scaling all distances by the same constant does not affect matching, thus with just λ we can form all possible matches that result from a linear combination of the two distances.

Propensity Equivalence

For large enough λ and small enough δ, the match generated from the caliper distance will be the same as the propensity match.

[6] Or $\max d_\mathbf{e}$ if the propensity distance is not $\in [0, 1]$.

[7] This is more of an issue when matching without replacement, but can still severely shrink the pool of available controls for many treated units.

Define λ^* as follows:

$$\lambda^* = \max_{i,j,k} \Bigg\{ \left| \frac{d_\Sigma(\mathbf{x}_i, \mathbf{x}_j) - d_\Sigma(\mathbf{x}_i, \mathbf{x}_k)}{d_\mathbf{e}(\mathbf{x}_i, \mathbf{x}_j) - d_\mathbf{e}(\mathbf{x}_i, \mathbf{x}_k)} \right| \\ \text{s.t. } d_\mathbf{e}(\mathbf{x}_i, \mathbf{x}_j) \neq d_\mathbf{e}(\mathbf{x}_i, \mathbf{x}_k) \;\&\; T_i = 1,\; T_j = T_k = 0 \Bigg\} \tag{4.13}$$

Lemma 4.2.1. *With $\delta = 0$ and $\lambda > \lambda^*$, the caliper distance $d_{\delta,\lambda}$ will reproduce propensity matches when matching **with replacement**, if a unique propensity match exists*[8].

Proof. For matching with replacement, we need only prove that each treated unit retains the same nearest neighbor. Let's relax notation and assume $d(i,j) = d(\mathbf{x}_i, \mathbf{x}_j)$.

WLOG, we're searching for a match for the first unit. Assume unit l is the unique minimiser of the propensity distance, $l = \text{argmin}_j d_\mathbf{e}(1,j)$.

For any other unit k, $d_{\delta=0,\lambda}(1,k) = d_\Sigma(1,k) + \lambda d_\mathbf{e}(1,k)$. Thus the difference in caliper distances is:

$$\begin{aligned} & d_{0,\lambda}(1,k) - d_{0,\lambda}(1,l) \\ &= d_\Sigma(1,k) - d_\Sigma(1,l) + \lambda d_\mathbf{e}(1,k) - \lambda d_\mathbf{e}(1,l) \\ &> d_\Sigma(1,k) - d_\Sigma(1,l) + \lambda^* \big(d_\mathbf{e}(1,k) - d_\mathbf{e}(1,l)\big) \\ &= \left(\lambda^* + \frac{d_\Sigma(1,k) - d_\Sigma(1,l)}{d_\mathbf{e}(1,k) - d_\mathbf{e}(1,l)}\right) \big(d_\mathbf{e}(1,k) - d_\mathbf{e}(1,l)\big) \end{aligned}$$

The fraction in the final term is bounded from below by $-\lambda^*$ by definition of λ^*. Thus we have $d_{0,\lambda}(1,k) > d_{0,\lambda}(1,l)$. □

λ^* is not the smallest value that satisfies the above, but we are not overly concerned about minimum bounds, since it's easy to find a bound that works in practice, and we only need one finite bound, not the smallest.

Lemma 4.2.2. *With $\delta = 0$, $\exists\ \lambda'$ such that for all $\lambda > \lambda'$, the caliper distance $d_{\delta,\lambda}$ will re-*

[8] That is, for any treated unit i, $min_j d_\mathbf{e}(\mathbf{x}_i, \mathbf{x}_j)$ has a unique solution j with $T_j = 0$.

produce propensity matches when matching ***optimally****, if a unique propensity match exists.*

Proof. Let $D^{\delta,\lambda}$ be the resulting distance matrix from using caliper distances, and similarly let $D^{\mathbf{e}}$ and D^{Σ} be the propensity and Mahalanobis distance matrices respectively.

With $\delta = 0$, for any match $\boldsymbol{M}$ we can write:

$$\sum_{i=1}^{N_t} D^{0,\lambda}_{\boldsymbol{M}_i} = \sum_{i=1}^{N_t} D^{\Sigma}_{\boldsymbol{M}_i} + \lambda \sum_{i=1}^{N_t} D^{\mathbf{e}}_{\boldsymbol{M}_i} \tag{4.14}$$

Which we can write as $D^{0,\lambda}(\boldsymbol{M}) = D^{\Sigma}(\boldsymbol{M}) + \lambda D^{\mathbf{e}}(\boldsymbol{M})$ with our notation from equation 4.3.

Let $\boldsymbol{M}^{\mathbf{e}}$ be the optimal propensity match, i.e. the solution to equation 4.4 with the propensity distance matrix. Let $\mathcal{T}_2$ be the minimum total propensity distance over all matches that are not $\boldsymbol{M}^{\mathbf{e}}$, i.e. $\mathcal{T}_2 = \min_{\boldsymbol{M}} \{D^{\mathbf{e}}(\boldsymbol{M}) : \ \boldsymbol{M} \neq \boldsymbol{M}^{\mathbf{e}}\}$. We can use:

$$\lambda^{\text{opt}} = \frac{N_t \max_{ij} D^{\Sigma}_{ij}}{\mathcal{T}_2 - D^{\mathbf{e}}(\boldsymbol{M}^{\mathbf{e}})} \tag{4.15}$$

as an upper bound for λ'.

Let $\boldsymbol{M}^{0,\lambda}$ be the optimal caliper match. If $\boldsymbol{M}^{\mathbf{e}} \neq \boldsymbol{M}^{0,\lambda}$, then $D^{\mathbf{e}}(\boldsymbol{M}^{0,\lambda}) > D^{\mathbf{e}}(\boldsymbol{M}^{\mathbf{e}})$, since the propensity solution is unique, and $D^{0,\lambda}(\boldsymbol{M}^{0,\lambda}) \leq D^{0,\lambda}(\boldsymbol{M}^{\mathbf{e}})$. Rearranging, this implies:

$$\lambda \leq \frac{D^{\Sigma}(\boldsymbol{M}^{\mathbf{e}}) - D^{\Sigma}(\boldsymbol{M}^{0,\lambda})}{D^{\mathbf{e}}(\boldsymbol{M}^{0,\lambda}) - D^{\mathbf{e}}(\boldsymbol{M}^{\mathbf{e}})} \tag{4.16}$$

The denominator is lower bounded by $\mathcal{T}_2 - D^{\mathbf{e}}(\boldsymbol{M}^{\mathbf{e}})$, and the numerator is upper bounded by N_t times the worst Mahalanobis distance, thus we get $\lambda \leq \lambda^{\text{opt}}$. □

To repeat, in practice values of λ that cause caliper matching to reproduce propensity matching are generally much lower than the required theoretical values, and can be found by any sensible search very quickly, especially since we don't need the minimum. Also δ does not need to be precisely zero to recreate propensity matches: for example setting it to

be smaller than any pairwise difference would work.

For non-unique propensity solutions, the caliper match will either resolve the ties by selecting among ties using Mahalanobis distance, or will produce the same matches stochastically if the issue of ties remains.

4.3 Evaluating Matches

The point of matching is to produce comparable sets of units, i.e. balanced. As Ho et al. (2007) write, we should try as many matching solutions as possible and select the best one. Of course, we cannot use the outcome to decide our match.

When matching, we want the resulting treated and control sets to be similar in their covariates: we would like the multivariate distribution of the covariates to be as close as possible in the treated and control groups. In general, it is not easy to evaluate this criterion.

4.3.1 Sets of Matches to Evaluate

In theory, we should actually try all the matching solutions. "All" will depend on what type of matching we're doing. Sticking with the pair matching framework, this gives $\frac{N_c!}{(N_c - N_t)!}$ optimal matches, and an even worse $N_t^{N_c}$ matches with replacement. Even in tiny matching problems, this is infeasible[9].

Instead, let's assume we've produced a subset of matches of size n, $\boldsymbol{\mathcal{M}} = \{\boldsymbol{M}_1, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_n\}$. We want to select a match to use for our analysis. We need criteria for this selection.

4.3.2 Relationship to Randomised Controlled Trials

Randomised controlled trials (RCTs) give the same distribution in the two groups, although of course they don't guarantee that the empirical distributions are identical. To force even greater balance in an RCT, stratification or blocking (Bernstein 1927) can be employed,

[9] With just fifty treated and fifty control units, there would be more matches than estimated atoms in the universe. For the "easier" number with optimal matches, we go above the atom estimate with sixty of each.

where units are first divided into homogeneous strata and then randomised into the arms of the study. This can reduce the variance of the treatment effect. RCTs break the association between treatment and outcome, as discussed in section 1.1.5. RCTs also have a huge extra advantage: they on average balance all unmeasured variables too.

For observational studies, we immediately lose the advantage of balancing unmeasured variables. However, we can do even better than RCTs in terms of balancing the measured covariates: instead of balancing in distribution or expectation, we can balance in-sample. Note that this can in fact make bias due to unmeasured variables even worse.

4.3.3 Covariate Balance

Most straightforwardly, our treated units and our control units should not differ greatly on any given covariate, i.e. we check that it is balanced between the two groups, or equivalently, balanced in the pairs[10]. A common metric to check is the mean difference between the two groups. That is, for a given match $\boldsymbol{M}$, for the jth covariate we are interested in:

$$\overline{b(\boldsymbol{M})}^{(j)} := \frac{1}{N_t} \sum_{i=1}^{N_t} \mathbf{x}^{(j)}_{\boldsymbol{M}_{i,1}} - \mathbf{x}^{(j)}_{\boldsymbol{M}_{i,2}} \tag{4.17}$$

In English, we calculate all the pairwise differences for each covariate from our match, and average them.

A simple univariate metric is then $\sum_j \left(\overline{b(\boldsymbol{M})}^{(j)}\right)^2$. Just like with Mahalanobis matching, an issue with this is that we may not want to weigh all variables equally. As in Gu and Rosenbaum (1993), this is rectified by scaling the covariate mean differences the same way as when defining the Mahalanobis distance. Let $\boldsymbol{b}$ be the vector of $\overline{b(\boldsymbol{M})}^{(j)}$ values. We can define the Mahalanobis imbalance Δ_Σ:

$$\Delta_\Sigma := \boldsymbol{b}' \Sigma^{-1} \boldsymbol{b} \tag{4.18}$$

[10]Not quite equivalent if e.g. using matching with replacement. In matches with repeated controls, or repeated treatments, we weight the covariates according to how many times each unit was used.

If using weights as in section 4.2.1, we can modify the above in just the same way that equation 4.5 becomes equation 4.7.

4.3.4 Multivariate Balance

An issue with the above is that the joint distribution of covariates can be imbalanced while each individual covariate j is balanced between the groups, i.e. the covariates are balanced marginally. If tall beer-drinkers and short wine-drinkers are our treated units, and short beer-drinkers and tall wine-drinkers are our controls, our marginal balance in a match could be perfect while the two groups are very different; any true interaction effect of the two variables on the outcome could severely bias our results, even with no unmeasured confounders.

Many methods have been proposed to more directly assess the multivariate balance between the two groups. Heller, Rosenbaum, and Small (2010) use Rosenbaum's cross match test (Rosenbaum 2005) to test multivariate balance. This test is based on forming a non-bipartite match, i.e. ignoring the treatment assignment and forming matches, and comparing the resulting match to the group assignments. Chen and Small (2016) propose a version of graph edge-tests that maintain power in observational studies.

The method most similar to what we will propose is the classification permutation test from Gagnon-Bartsch and Shem-Tov (2016). We will describe their method further in section 4.3.6.

4.3.5 Relating Multivariate Equivalence to Probability

Multivariate balance would imply the post-match density f_m of any vector of covariates $\mathbf{x}$ is the same for treated and control units, i.e.:

$$f_m(\mathbf{x} \mid T = 1) = f_m(\mathbf{x} \mid T = 0) \tag{4.19}$$

Note that the post-match density is a function of the matching procedure, and often bears little resemblance to the pre-match joint density of $\mathbf{x}$ and T, i.e. $f_1(\mathbf{x})$ and $f_1(\mathbf{x})$ from previous sections.

This is hard to directly test with high power unless the dimension of $\mathbf{x}$ is very small. Bayes' theorem relates these densities to the probability of post-match treatment. Let $\mathbf{P}_m$ refer to post-match probability:

$$\begin{aligned}\mathbf{P}_m(T=1 \mid \mathbf{x}) =& \frac{f_m(\mathbf{x} \mid T=1)\mathbf{P}_m(T=1)}{f_m(\mathbf{x})} \\ \mathbf{P}_m(T=0 \mid \mathbf{x}) =& \frac{f_m(\mathbf{x} \mid T=0)\mathbf{P}_m(T=0)}{f_m(\mathbf{x})}\end{aligned} \tag{4.20}$$

Therefore:

$$\begin{gathered} f_m(\mathbf{x} \mid T=1) = f_m(\mathbf{x} \mid T=0) \\ \iff \\ \frac{\mathbf{P}_m(T=1 \mid \mathbf{x})}{\mathbf{P}_m(T=1)} = \frac{\mathbf{P}_m(T=0 \mid \mathbf{x})}{\mathbf{P}_m(T=0)} \end{gathered} \tag{4.21}$$

Since $\mathbf{P}_m(T=1 \mid \mathbf{x}) = 1 - \mathbf{P}_m(T=0 \mid \mathbf{x})$ and $\mathbf{P}_m(T=1) = 1 - \mathbf{P}_m(T=0)$, this is equivalent to:

$$\mathbf{P}_m(T=1 \mid \mathbf{x}) = \mathbf{P}_m(T=1) \tag{4.22}$$

That is, knowing $\mathbf{x}$ doesn't tell us anything about treatment status.

Pair Matching and Multivariate Equivalence

If our outcome analysis is on the level of matched pairs, we are interested in a pair-level density match. Let g be the post-match density of pairs of covariate vectors. This means we want:

$$g_m(\mathbf{x}_i, \mathbf{x}_j \mid T_i=1, T_j=0) = g_m(\mathbf{x}_i, \mathbf{x}_j \mid T_i=0, T_j=1) \tag{4.23}$$

There is no inherent ordering in the pairs, thus we must have $g_m(\mathbf{x}_i, \mathbf{x}_j \mid T_i = t_i, T_j = t_j) = g_m(\mathbf{x}_j, \mathbf{x}_i \mid T_j = t_j, T_i = t_i)$ by symmetry. Combining this with the above equation, we get:

$$g_m(\mathbf{x}_i, \mathbf{x}_j \mid T_i = t_i, T_j = t_j) = g_m(\mathbf{x}_j, \mathbf{x}_i \mid T_j = t_i, T_i = t_j) \quad \forall t_i, t_j \tag{4.24}$$

We'll call this joint balance.

Since we form matched pairs, we can assume $\mathbf{P}_m(T_i = 1, T_j = 1) = \mathbf{P}_m(T_i = 0, T_j = 0)$. Bayes' theorem gives us the equivalent of 4.22:

$$\mathbf{P}_m(T_i = t_i, T_j = t_j \mid \mathbf{x}_i, \mathbf{x}_j) = \mathbf{P}_m(T_i = t_i, T_j = t_j) \tag{4.25}$$

Essentially, if we're shown a pair $(\mathbf{x}_i, \mathbf{x}_j)$, we want to have no idea which of the two belongs to a treated unit, and which to a control.

Joint balance implies equation 4.19, i.e. distributional balance, in situations where both exist - i.e. if we make a pair match, we can also ignore the pairings and analyse the distributions of the two groups.

Lemma 4.3.1. *Joint balance implies distributional balance, i.e. equation 4.24 implies equation 4.19.*

Proof.

$$\begin{gathered} f_m(\mathbf{x} \mid 1) = \\ \int_{\boldsymbol{\nu}} g_m(\mathbf{x}, \boldsymbol{\nu} \mid (1,0))\, \mathrm{d}\boldsymbol{\nu} = \\ \int_{\boldsymbol{\nu}} g_m(\mathbf{x}, \boldsymbol{\nu} \mid (0,1))\, \mathrm{d}\boldsymbol{\nu} = \\ f_m(\mathbf{x} \mid 0) \end{gathered} \tag{4.26}$$

□

Joint balance is stricter than equation 4.19. Say we wanted to match on a single variable $\boldsymbol{x}$ that took three values, $1, 2$ and 3, and this variable had the same distribution in both

treatment and control groups. If we matched treated units with $x = 1$ to control units with $x = 2$, treated units with $x = 2$ to control units with $x = 3$ and finally treated units with $x = 3$ to control units with $x = 1$, equations 4.19 and 4.22 would be satisfied. However, we'd have:

$$\begin{aligned} \mathbf{P}_m(T_i = 1, T_j = 0 \mid x_i = 1, x_j = 2) &= 1 \\ \mathbf{P}_m(T_i = 1, T_j = 0 \mid x_i = 1, x_j = 3) &= 0 \end{aligned} \tag{4.27}$$

And neither would be equal to the unconditional value of $\mathbf{P}_m(T_i = 1, T_j = 0)$.

While it's clear in this toy example that we've simply matched poorly, it is much less clear in large multivariate matches. Even in univariate cases where our analysis is not on the level of pairs, checking pairwise density equivalence can have greater power against the alternative of non-equal distributions than checking the non-pairwise density. We provide an example in the next section.

Testing Pair Differences

Let $X \sim$ Unif(0,1). Say $\mathbf{P}(T = 1 \mid X = x) = x$, i.e. the pre-match probability (i.e. population, or real, probability, not $\mathbf{P}_m$) of treatment. Let Y be a function of T and X plus noise. A correctly specified propensity match or a Mahalanobis match will both match based on $|x_i - x_j|$. On average, this will result in the matched pairs having slightly larger values of x in the treated units than the control units, but the two distributions will look similar.

For a specific simulation of the above: set $N = 200$. Let $Y = T + 2x + 4x^2 + \mathcal{N}(0, 1)$. We perform a match with replacement[11]. We ran $50,000$ simulations.

The resulting estimate is biased, with a mean of about 1.09. Only the variance calculation differs between paired estimation and unpaired estimation, so whether or not we like pairwise differences in the outcome, we have a bias. A Kolmogorov-Smirnov has 37% power to detect

[11] The same issue affects optimal matching, the power calculations are just more awkward to perform.

the discrepancy $f_m(x \mid T = 1) \neq f_m(x \mid T = 0)$, i.e. that x is not distributed equally in the treatment and control sets. Testing that the pairwise differences are distributed around zero has 71% power[12], which is a simple test of a null implied by 4.24, as we'll see in the next section.

Distribution of Difference

The equality in joint distribution, equation 4.24, implies the post-match density of the difference in covariates within a matched pair is symmetric. Let h_m be the density of the differences.

$$h_m(\mathbf{x}_i - \mathbf{x}_j \mid T_i = 1, T_j = 0) = h_m(\mathbf{x}_i - \mathbf{x}_j \mid T_i = 0, T_j = 1) \tag{4.28}$$

Like in section 4.3.5, due to a lack of inherent ordering, we have $h_m(\boldsymbol{\nu} \mid t_i, t_j) = h_m(-\boldsymbol{\nu} \mid t_j, t_i)$ by symmetry.

Similar to equation 4.24, it can be more illuminating to rewrite this:

$$h_m(\mathbf{x}_i - \mathbf{x}_j \mid T_i = t_i, T_j = t_j) = h_m(\mathbf{x}_j - \mathbf{x}_i \mid T_j = t_i, T_i = t_j) \quad \forall t_i, t_j \tag{4.29}$$

In words, the distribution of the difference in covariates does not depend on the order of subtraction.

Lemma 4.3.2. *Joint balance implies difference symmetry, i.e. equation 4.24 implies equation 4.29.*

Proof. Let $\mathbf{z}$ represent difference in covariates, so that $\mathbf{z}_{i,j} = \mathbf{x}_i - \mathbf{x}_j$.

[12] 71% for a straight-up $t-$test, approx 58% for a signed rank test. Interestingly, the unpaired $t-$test has extremely low power, while the Kolmogorov-Smirnov test on the differences, i.e. testing all differences in any order, against the opposite order, has poor power.

$$
\begin{aligned}
&h_m(\mathbf{z}_{i,j} \mid (t_i, t_j)) = \\
&\int_{\boldsymbol{\nu}} g_m(\boldsymbol{\nu}, \boldsymbol{\nu} - \mathbf{z}_{i,j} \mid (t_i, t_j))\, \mathrm{d}\boldsymbol{\nu} = \\
&\int_{\boldsymbol{\nu}} g_m(\boldsymbol{\nu} - \mathbf{z}_{i,j}, \boldsymbol{\nu} \mid (t_i, t_j))\, \mathrm{d}\boldsymbol{\nu} = \\
&h_m(-\mathbf{z}_{i,j} \mid (t_i, t_j))
\end{aligned}
\tag{4.30}
$$

□

We discuss in section 4.3.9 that the reverse is not true: the above equation 4.29 does not imply equality in joint distribution.

4.3.6 Predicting Treatment

More than the true post-match density f_m and the true probabilities $\mathbf{P}_m(T \mid \mathbf{x})$, we would like the equivalent in section 4.3.5 to be true for the empirical density $\tilde{f}_m$, and the in-sample probabilities $\tilde{\mathbf{P}}_m$. That is, we want to test if:

$$
\tilde{\mathbf{P}}_m(T = 1 \mid \mathbf{x}) = \tilde{\mathbf{P}}_m(T = 1) \tag{4.31}
$$

The right-hand side is just the proportion of units that are treated, after potentially having dropped some units from our analysis[13].

Another way to say this: we don't want to be able to predict which units in our match are the treated units.

Testing equation 4.31 is exactly what the classification permutation test, or CPT[14], does. Assume we have a match $\boldsymbol{M}$, and let $N_{\boldsymbol{M}} \leq N$ be the number of units in our match[15]. With a small abuse of notation, let (i) index the ith unit in the match[16]. Then the CPT

[13] So far we have only dropped control units, but we will extend this in section 4.7.

[14] Again, from Gagnon-Bartsch and Shem-Tov (2016)

[15] In optimal pair matching, using all treated units, we'll have $N_{\boldsymbol{M}} = 2N_t$. When matching with replacement, $N_t + 1 \leq N_{\boldsymbol{M}} \leq 2N_t$.

[16] Without duplicated controls, we would write $\boldsymbol{M}_j = (T_{(j)}, T_{(N_t+j)})$, i.e. the ith unit from our match is the ith treated unit if $i \leq N_t$, and the $i - N_t$th control unit if $i > N_t$. We use the same notation if we allow

method is:

CPT Method:

1. Train any classifier to predict T from $\mathbf{x}$, using all data from units remaining after matching

2. Predict the class $\hat{T} \in \{0, 1\}$ for every unit

3. Record the accuracy:

$$S := \frac{1}{N_{\boldsymbol{M}}} \sum_{i=1}^{N_{\boldsymbol{M}}} \mathbf{1}(\hat{T}_{(i)} = T_{(i)}) \tag{4.32}$$

   S is the test statistic.

4. Use permutation inference to evaluate S: Permute $\mathbf{T}$ some large number L times, train the classifier for each permutation, retain the vector of permutation accuracies $\boldsymbol{S}^\star$.

5. Calculate the p-value as the proportion of elements of $\boldsymbol{S}^\star$ greater than or equal to S:

$$\mathbf{p}_{\text{cpt}} = \frac{1}{L} \sum_{l=1}^{L} \mathbf{1}(S \geq S_l^\star) \tag{4.33}$$

The last two steps ensure the p-value is valid. We could instead use something that looks like the correct distribution to calculate the p-value: the binomial distribution with $n = N_{\boldsymbol{M}}$ and $p = \overline{\boldsymbol{T}}$, the proportion of treated in the match. The permutation method provides an advantage: the classier is allowed to overfit the data - we can use in-sample fits. Secondly, if we use a different test statistic, we don't have to know its full null distribution[17].

4.3.7 Brier Testing Procedure: Marginal

Our main testing procedure for a given match resembles the CPT. Instead of just testing a match, we use this procedure for match selection. For a given match $\boldsymbol{M}$, the data is $(\mathbf{x}_j, T_j)$

duplicate matches, but we only count each unit once.

[17] And even if we do, we could be spared analytical evaluation

for all $j \in \boldsymbol{M}$.

Marginal Brier Testing:

1. Pick any classifier. Split the data into K folds. For each fold k:

   a Train the classifier to predict T from $\mathbf{x}$, using all data not in fold k

   b Predict the treatment probability $\hat{T} \in [0, 1]$ for every unit in fold k

2. Record the Brier score:

$$B := \frac{1}{N_{\boldsymbol{M}}} \sum_{i=1}^{N_{\boldsymbol{M}}} \left(\hat{T}_{(i)} - T_{(i)}\right)^2 \tag{4.34}$$

   B is our test statistic.

3. Use permutation inference to evaluate B: Permute $\mathbf{T}$ some large number L times, train the classifier for each permutation, retain the vector of permutation accuracies $\boldsymbol{B}^\star$.

4. Calculate the p-value as the proportion of elements of $\boldsymbol{B}^\star$ more extreme than B, in either direction[18]:

$$\mathbf{p}_{\mathrm{B}} = \frac{2}{L} \min \left\{ \sum_{l=1}^{L} \mathbf{1}(B \leq B_l^\star),\ \sum_{l=1}^{L} \mathbf{1}(B \geq B_l^\star) \right\} \tag{4.35}$$

   Also calculate the one-sided version $\mathbf{p}_{\mathrm{B}}^{\mathrm{L}}$, which only counts how many permutation Brier scores are less than our given score B.

$$\mathbf{p}_{\mathrm{B}}^{\mathrm{L}} = \frac{1}{L} \sum_{l=1}^{L} \mathbf{1}(B \geq B_l^\star) \tag{4.36}$$

To explain the differences between this procedure and the CPT: step one forces "out of sample" prediction. We don't just want valid p values, we want to use these scores to select

[18] This corresponds to typical two-sided p-values.

between matches. The best in-sampler performer is much less interesting than the best out-of-sample performer[19]. For step two: Brier scoring is a strict proper scoring rule, unlike misclassification rate, which is non-strict. Further, misclassification rate is noisy, or unstable (Buja, Stuetzle, and Shen 2005). One could of course substitute log-loss, or any other scoring rule.

We use two-sided p-values to avoid declaring success with an overfit match: if we somehow are very consistently picking the wrong unit as the treated, we may be just rewarding an unlucky data split. This would happen even with perfect balance: selecting the permutation match with the highest p-value would also select for a lucky, or unlucky depending on your perspective, data split.

$B, \mathbf{p}_{\mathrm{B}}$ and $\mathbf{p}_{\mathrm{B}}^{\mathrm{L}}$ are functions of the match. Write $B(\boldsymbol{M})$ for the Brier score for match $\boldsymbol{M}$, and similarly $\mathbf{p}_{\mathrm{B}}(\boldsymbol{M})$ and $\mathbf{p}_{\mathrm{B}}^{\mathrm{L}}(\boldsymbol{M})$

Finally, as we'll expand on in the next section, we generally do not train and predict on $(\mathbf{x}, T)$, but rather on pairs in our match.

4.3.8 Predicting Treatment From Pairs

Just like in section 4.3.5, we might care specifically about the equality of distributions between the covariates of the actual matched pairs, rather than the whole groups.

This doesn't require a huge change to the Brier testing method detailed previously. The data is now a quadruple $(\mathbf{x}_i, T_i, \mathbf{x}_j, T_j)$. Our predictor should predict (T_i, T_j) from $(\mathbf{x}_i, \mathbf{x}_j)$, while generally conditioning on $T_i + T_j = 1$, i.e. that exactly one unit is treated.

To be specific, for each row of $\boldsymbol{M}$, we try to predict which of $\boldsymbol{M}_{i,1}$ and $\boldsymbol{M}_{i,2}$ is the treated unit, using $(\mathbf{x}_{\boldsymbol{M}_{i,1}}, \mathbf{x}_{\boldsymbol{M}_{i,2}})$.

This leads to our general Brier testing procedure, which looks similar to our Marginal procedure:

[19] Really we mean the best cross-validated performer

Brier Testing:

1. Pick any classifier. Split the data into K folds. For each fold k:

   a Train the classifier to predict (T_a, T_b) from $(\mathbf{x}_a, \mathbf{x}_b)$, the pairs in our match, using all data not in fold k

   b Predict the treatment probability $\hat{T} \in [0, 1]$ for every treated unit in fold k. That is, feed the quadruple $(\mathbf{x}_i, T_i, \mathbf{x}_j, T_j)$ for all pairs from the match in fold k, and record the probability of the actual treated unit being treated.

2. Record the Brier score, averaging over treated units:

$$B := \frac{1}{N_t} \sum_{i=1}^{N_t} \left(\hat{T}_{(i)} - T_{(i)}\right)^2 \tag{4.37}$$

   B is our test statistic.

3. Use permutation inference to evaluate B: Permute the data (treatment and covariate quadruples) some large number L times, train the classifier for each permutation, retain the vector of permutation accuracies $\boldsymbol{B}^\star$.

4. Calculate the p-value as the proportion of elements of $\boldsymbol{B}^\star$ more extreme than B, in either direction[20]:

$$\mathbf{p}_{\mathrm{B}} = \frac{2}{L} \min \left\{ \sum_{l=1}^{L} \mathbf{1}(B \leq B_l^\star),\ \sum_{l=1}^{L} \mathbf{1}(B \geq B_l^\star) \right\} \tag{4.38}$$

   Also calculate the one-sided version $\mathbf{p}_{\mathrm{B}}^{\mathrm{L}}$, which only counts how many permutation Briers are less than our given score B.

$$\mathbf{p}_{\mathrm{B}}^{\mathrm{L}} = \frac{1}{L} \sum_{l=1}^{L} \mathbf{1}(B \geq B_l^\star) \tag{4.39}$$

[20] This corresponds to typical two-sided p-values.

Row Switching

Naturally there is a trivial solution: we already set up our matrix $\boldsymbol{M}$ so that the first column contains the treated units, thus unit $\boldsymbol{M}_{i,1}$ is the treated unit - predict the first unit is the treated and we'll do perfectly. This tests absolutely nothing - it's equivalent to reordering a group of people by height and subsequently trying to guess the order.

One easy solution to this triviality is to randomly switch the pairs $(\mathbf{x}_i, \mathbf{x}_j)$ prior to training, and similarly the treatment indicators. That is, with probability $1/2$, the vector $(\mathbf{x}_i, \mathbf{x}_j, 1, 0)$[21] is part of the training set; else the vector $(\mathbf{x}_j, \mathbf{x}_1, 0, 1)$ is part of it.

Don't Switch, and Do

Another solution is to add both vectors above to the training set. This gives misleading in-sample fits, but performs better out of sample than random switching. This is our preferred solution.

To be specific, let $\boldsymbol{X}_{\text{Tr}}$ be the matrix containing our match's treated units' covariates: the ith row of $\boldsymbol{X}_{\text{Tr}}$ is $\mathbf{x}_{\boldsymbol{M}_{i,1}}$. Similarly, let $\boldsymbol{X}_{\text{Ctrl}}$ be the matrix of control covariates from our match, with ith row equal to $\mathbf{x}_{\boldsymbol{M}_{i,2}}$. The predictor matrix for our method is then:

$$\boldsymbol{X}_{\text{both}} := \begin{bmatrix} \boldsymbol{X}_{\text{Tr}} & \boldsymbol{X}_{\text{Ctrl}} \\ \boldsymbol{X}_{\text{Ctrl}} & \boldsymbol{X}_{\text{Tr}} \end{bmatrix} \tag{4.40}$$

$\boldsymbol{X}_{\text{both}}$ will have $2N_t$[22]rows, and twice the number of columns as $\mathbf{X}$.

The outcome to be predicted is then:

$$\boldsymbol{T}_{\text{both}} := \begin{bmatrix} \boldsymbol{T}_{\text{Tr}} & \boldsymbol{T}_{\text{Ctrl}} \\ \boldsymbol{T}_{\text{Ctrl}} & \boldsymbol{T}_{\text{Tr}} \end{bmatrix} \tag{4.41}$$

[21] WLOG this assumes $T_i = 1$, $T_j = 0$.
[22] Of course we subset this matrix when doing our k-fold out of sample fitting.

Where $\boldsymbol{T}_{\text{Tr}} = [1, 1, \ldots, 1]'$ and $\boldsymbol{T}_{\text{Ctrl}} = [0, 0, \ldots, 0]'$, both of length N_t. Predicting one of $(1, 0)$ and $(0, 1)$ is not really a two dimensional outcome vector, because it's fully dependent: we can instead just predict the first element, and thus redefine $\boldsymbol{T}_{\text{both}}$ as $[\boldsymbol{T}'_{\text{Tr}}, \boldsymbol{T}'_{\text{Ctrl}}]'$.

When doing any sort of cross-validation or equivalent, we do not sample rows randomly from $\boldsymbol{X}_{\text{both}}$: the point of the switch is to force the classifier to learn that switching the order of the inputs along with the outputs should give the same output. Thus we sample indices, and use those to build $\boldsymbol{X}^{\text{train}}_{\text{both}}$ and $\boldsymbol{X}^{\text{test}}_{\text{both}}$. That is, if the ith row of $\boldsymbol{X}_{\text{both}}$ is included in the training set, then so should the $i + N^t$th row.

Unpaired and Differences Prediction

Even if we want to predict on the pair scale, the previous methods double the number of parameters. If the number of covariates is large relative to N_t, then twice the number of covariates is even larger. This could make pair prediction either infeasible, or just noisier than unpaired prediction, i.e. prediction in section 4.3.6.

We can modify unpaired predictions in one simple way. For a matched pair (T_i, T_j) we have $\hat{T}_i$ and $\hat{T}_j$; we can condition on the sum to improve prediction, i.e. use $\frac{\hat{T}_i}{\hat{T}_i + \hat{T}_j}$ as our fit for T_i.

Alternatively, as mentioned in section 4.3.9, and as done in the special case of logistic regression in the following section, we can analysis the differences in covariates in the matched pairs: $\mathbf{x}_i - \mathbf{x}_j$ where (i, j) is a pair in our match.

Predicting using the differences can be done in a similar fashion to the pair-wise methods. We can randomly order the subtraction, remembering to order the treatment vector for prediction accordingly, or simply follow the above section and put both versions of the differences into the classifier.

Special Case: Logistic Regression

The method from 4.3.8 is unnecessary if our classier is logistic regression, and we have no interaction terms[23].

There are two ways to see this. Firstly, if our model predicts some probability p for the vector $(\mathbf{x}_i, \mathbf{x}_j)$, it should predict $1 - p$ if we switch the order of units around, $(\mathbf{x}_j, \mathbf{x}_i)$, by symmetry.

Let $\boldsymbol{\beta}_{\text{both}}$ be the coefficient vector in logistic regression for the matrix $\boldsymbol{X}_{\text{both}}$, when predicting the treatment assignment of the first unit, i.e when the ith row of $\boldsymbol{X}_{\text{both}}$ is $(\mathbf{x}_a, \mathbf{x}_b)$, we have:

$$\text{logit } \mathbf{P}(T_i = 1, T_j = 0 \mid (\mathbf{x}_a, \mathbf{x}_b)) = \boldsymbol{\beta}'_{\text{both}}(\mathbf{x}_a, \mathbf{x}_b) \tag{4.42}$$

Split the coefficient vector into two, so that $\boldsymbol{\beta}_{\text{both}} = [\boldsymbol{\beta}'_1,\ \boldsymbol{\beta}'_2]'$. Then the probability of the first unit being the treated unit, i.e. unit a in the pair of matched units (a, b), is $\text{logit}^{-1}(\boldsymbol{\beta}'_1\mathbf{x}_a + \boldsymbol{\beta}'_2\mathbf{x}_b)$[24], and thus the probability that b is the treated unit is $1 - \text{logit}^{-1}(\boldsymbol{\beta}'_1\mathbf{x}_a + \boldsymbol{\beta}'_2\mathbf{x}_b)$.

If we switch the order of the coefficient vectors around, we get that the probability that unit b is the treated in the pair (b, a) is $\text{logit}^{-1}(\boldsymbol{\beta}'_1\mathbf{x}_b + \boldsymbol{\beta}'_2\mathbf{x}_a)$. For these to be equal, we

[23] Of course we usually do care about interaction terms, hence why using plain logistic is not advised.

[24] $\text{logit}(p) = \frac{p}{1-p}$, so $\text{logit}^{-1}(x) = \frac{e^x}{1+e^x}$, or $\frac{1}{1+e^{-x}}$

need:

$$\begin{gathered}
1 - \text{logit}^{-1}(\boldsymbol{\beta}_1'\mathbf{x}_a + \boldsymbol{\beta}_2'\mathbf{x}_b) = \text{logit}^{-1}(\boldsymbol{\beta}_1'\mathbf{x}_b + \boldsymbol{\beta}_2'\mathbf{x}_a) \\
\iff \\
\frac{1}{1 + \exp(\boldsymbol{\beta}_1'\mathbf{x}_a + \boldsymbol{\beta}_2'\mathbf{x}_b)} = \frac{\exp(\boldsymbol{\beta}_1'\mathbf{x}_b + \boldsymbol{\beta}_2'\mathbf{x}_a)}{1 + \exp(\boldsymbol{\beta}_1'\mathbf{x}_b + \boldsymbol{\beta}_2'\mathbf{x}_a)} \\
\iff \\
\exp(\boldsymbol{\beta}_1'\mathbf{x}_a + \boldsymbol{\beta}_2'\mathbf{x}_b + \boldsymbol{\beta}_1'\mathbf{x}_b + \boldsymbol{\beta}_2'\mathbf{x}_a) = 1 \\
\iff \\
(\boldsymbol{\beta}_1' + \boldsymbol{\beta}_2')(\mathbf{x}_a + \mathbf{x}_b) = 0
\end{gathered} \tag{4.43}$$

For this to be true for all $\mathbf{x}_a, \mathbf{x}_b$, we need:

$$\boldsymbol{\beta}_1 = -\boldsymbol{\beta}_2 \tag{4.44}$$

To prove this:

Lemma 4.3.3. *The likelihood of the described logistic method will be maximised at* $\boldsymbol{\beta}_1 = -\boldsymbol{\beta}_2$.

Proof. For any vector $[\boldsymbol{\beta}_1',\ \boldsymbol{\beta}_2']'$, we'll show that $[\boldsymbol{\beta}_\star', -\boldsymbol{\beta}_\star']'$ produces a higher log likelihood, where:

$$\boldsymbol{\beta}_\star = \frac{\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2}{2} \tag{4.45}$$

Let $\mathbf{x}_{t,i}$ be the ith row of $\boldsymbol{X}_{\text{Tr}}$, and similarly let $\mathbf{x}_{c,i}$ be the ith row of $\boldsymbol{X}_{\text{Ctrl}}$, thus $(\mathbf{x}_{t,i}, \mathbf{x}_{c,i})$ is the ith row of $\boldsymbol{X}_{\text{both}}$, and $(\mathbf{x}_{c,i}, \mathbf{x}_{t,i})$ is the $i + N_t$th row. The ith outcome is 1 and the $i + N_t$th outcome is 0, therefore the likelihood of the logistic method can be written:

$$\prod_{i=1}^{N_t} \text{logit}^{-1}(\boldsymbol{\beta}_1'\mathbf{x}_{t,i} + \boldsymbol{\beta}_2'\mathbf{x}_{c,i}) \times \prod_{i=1}^{N_t} \left(1 - \text{logit}^{-1}(\boldsymbol{\beta}_1'\mathbf{x}_{c,i} + \boldsymbol{\beta}_2'\mathbf{x}_{t,i})\right) \tag{4.46}$$

Let's focus on the pair of products for any i, i.e. the ith term of the first product and the ith term of the second. Let $e_1 = \exp(-\boldsymbol{\beta}_1'\mathbf{x}_{t,i} - \boldsymbol{\beta}_2'\mathbf{x}_{c,i})$ and let $e_2 = \exp(\boldsymbol{\beta}_1'\mathbf{x}_{c,i} + \boldsymbol{\beta}_2'\mathbf{x}_{t,i})$.

We can write the likelihood contribution as:

$$\frac{1}{1+e_1}\frac{1}{1+e_2} \tag{4.47}$$

Let's look at the linear terms when we use $\boldsymbol{\beta}_\star$ and $-\boldsymbol{\beta}_\star$. In the ith term of the first product, we get:

$$\begin{gathered}\boldsymbol{\beta}_\star'\mathbf{x}_{t,i} - \boldsymbol{\beta}_\star'\mathbf{x}_{c,i} = \\ \frac{\boldsymbol{\beta}_1' - \boldsymbol{\beta}_2'}{2}\mathbf{x}_{t,i} - \frac{\boldsymbol{\beta}_1' - \boldsymbol{\beta}_2'}{2}\mathbf{x}_{c,i} = \\ \frac{\boldsymbol{\beta}_1'\mathbf{x}_{t,i} + \boldsymbol{\beta}_2'\mathbf{x}_{c,i}}{2} - \frac{\boldsymbol{\beta}_1'\mathbf{x}_{c,i} + \boldsymbol{\beta}_2'\mathbf{x}_{t,i}}{2}\end{gathered} \tag{4.48}$$

Thus $\exp\big(-(\boldsymbol{\beta}_\star'\mathbf{x}_{t,i} - \boldsymbol{\beta}_\star'\mathbf{x}_{c,i})\big) = \sqrt{e_1 e_2}$. Similarly, for the ith term of the second product, we get exp(linear term) $= \sqrt{e_1 e_2}$[25]. Thus the contribution to the log-likelihood for the two terms is:

$$\frac{1}{1+\sqrt{e_1 e_2}}\frac{1}{1+\sqrt{e_1 e_2}} \tag{4.49}$$

Since $(\sqrt{e_1} - \sqrt{e_2})^2 \geq 0$, this product must be greater than equation 4.47. This inequality is strict unless $e_1 = e_2$, which leads us back to equation 4.43.

Thus given an MLE, we must have $\boldsymbol{\beta}_1 = -\boldsymbol{\beta}_2$, or we can strictly increase the likelihood[26]. □

But all this implies we'd get the same model by using the vector of coefficient differences $\mathbf{x}_i - \mathbf{x}_j$ instead of the pair vector; further, we don't need to do the switching, since this would produce the exact same fit as just the unswitched differences.

Rather oddly, this means we can fit a logistic regression with the outcome vector being all unity! In fact we can see this another way: as noted previously, $\frac{1}{1+e^{-x}} = 1 - \frac{1}{1+e^{x}}$. Thus for

[25] $\text{logit}^{-1}(x) = \frac{1}{1+e^{-x}}$, while $1 - \text{logit}^{-1}(x) = \frac{1}{1+e^{x}}$, hence why we care about exp(−) for the first term and exp(+) for the second.

[26] And indeed, if $\boldsymbol{\beta}_1 = -\boldsymbol{\beta}_2$, then $\boldsymbol{\beta}_\star = {}^1\!/_2\big(\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2\big) = {}^1\!/_2\big(\boldsymbol{\beta}_1 + \boldsymbol{\beta}_1\big) = \boldsymbol{\beta}_1$

any data matrix, changing the outcome bit from 1 to 0 or vice-versa, and also changing the sign of the corresponding row, has exactly no effect on the log-likelihood. Thus we can keep flipping all outcome bits until all are unity (or e.g. all zero), and we'll have the same fit.

Naturally this sounds intuitively wrong. But something that gets hidden in the above, or when taking purely differences: the intercept is completely transformed. In taking differences, there is no intercept anymore. In flipping labels, the intercept is in fact $2Y - 1$, where Y is the outcome vector. In both cases, the model is now trying to find the linear combination of inputs that's as large as possible[27]. There is no trivial solution issue: e.g. any linear dependence in the flipped version must have existed previously.

To reiterate: if we're regressing on the vectors of differences, we can simply not include an intercept, and use $Y \equiv 1$ as our outcome.

4.3.9 Pairwise Distribution is not the Difference Distribution

We could be tempted to always use the vector of covariate differences in our matched pairs. After all, if equation 4.24 holds, the distribution of differences must be symmetric, i.e. equation 4.29 holds, and thus any asymmetry that is predictive of treatment status indicates a deficient match.

However the reverse is not true: if the distribution of differences is symmetric, we may still be able to predict treatment status by not reducing to differences. Trivially this is true if the distributions for treated and control are already not the same: e.g. if our treated units have a univariate x that takes the values 0 and 3 equally, and are always matched to 1 and 2 respectively, the differences will be -1 and 1 in equal proportion, and thus differences will be symmetric and non-predictive[28], but any (nonlinear) analysis of the distribution of x would reveal the dependence.

[27] Where the loss function is the same as logistic.

[28] For predictive purposes, we'll see four distinct inputs: $(0,1), (1,0), (3,2), (2,3)$. The treatment outcomes will be $[1,0,1,0]$ and the differences $[-1,1,1,-1]$. Thus the differences and treatment outcomes are independent.

We can also construct examples where the density of the covariates is equal in both groups, the density of the differences is symmetric, and yet the density of the pairs is predictive. See appendix A.4 for an example. This example is somewhat contrived, and in reality it's unlikely we'd have both the difference distribution and the unpaired distribution be perfectly balanced while the joint is not. The main benefit is similar to section 4.3.5: in situations where all are violated, the joint distribution often has more power to detect the imbalance[29].

Pairwise Power

We've already discussed in the previous section and in section 4.3.5 that there are examples where pairwise, or joint, balance doesn't hold, but both the unpaired distributions and the difference distributions are blind to it.

The example in the appendix quickly shows that even if both equations 4.19 and 4.29 jointly hold, we still don't have guarantees about joint balance.

The pairwise distribution is asymptotically at least as powerful as the best of the other two methods. The following example makes this clear for the normal distribution.

Multivariate Normal Example

Say g_m is the multivariate normal distribution, i.e.:

$$X_i, X_j | T_{ij} = (1,0) \sim \mathcal{N}\left(\begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix}, \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma'_{ab} & \Sigma_b \end{pmatrix} \right) \tag{4.50}$$

By pair symmetry, we thus have $\mathcal{N}\left(\begin{pmatrix} \mu_b \\ \mu_a \end{pmatrix}, \begin{pmatrix} \Sigma_b & \Sigma'_{ab} \\ \Sigma_{ab} & \Sigma_a \end{pmatrix} \right)$ if we switch around which unit is treated, i.e. $T_{ij} = (0,1)$. This is not joint balance, but places restrictions on g_m.

[29] It must be noted that the example in section 4.3.5 used a metric based on the difference distribution - this was mostly for simplicity. Using our Brier testing framework with xgboost increases the power to approx 99%.

We can find f_m and h_m from this. The unpaired density is straight-forward:

$$\begin{aligned} X|T=1 &\sim \mathcal{N}(\mu_a,\ \Sigma_a) \\ X|T=0 &\sim \mathcal{N}(\mu_b,\ \Sigma_b) \end{aligned} \tag{4.51}$$

For the difference:

$$\begin{aligned} X_i - X_j|T_{ij}=(1,0) &\sim \mathcal{N}(\mu_a-\mu_b,\ \Sigma_a+\Sigma_b+\Sigma_{ab}+\Sigma'_{ab}) \\ X_i - X_j|T_{ij}=(0,1) &\sim \mathcal{N}(\mu_b-\mu_a,\ \Sigma_b+\Sigma_a+\Sigma'_{ab}+\Sigma_{ab}) \end{aligned} \tag{4.52}$$

From here, it's immediate that the difference distribution is only sensitive to differences in the means, while the unpaired is sensitive to differences in both the means and the variances. Neither will be sensitive to Σ_{ab}, the covariances between the two units.

In simulation, when $\mu_a \neq \mu_b$, the difference distribution is much more sensitive to this than the unpaired distribution, and the joint is on par with the difference distsribution. When the means are equal but $\Sigma_b \neq \Sigma_a$, the difference distribution finds nothing, while the joint outperforms the unpaired.

Finally, looking at Σ_{ab}: using the pair symmetry always present in g_m[30], we see that joint distribution is satisfied for $\mu_a = \mu_b$, $\Sigma_b = \Sigma_a$ and $\Sigma'_{ab} = \Sigma_{ab}$. Thus if $\Sigma'_{ab} \neq \Sigma_{ab}$ is a violation of interest in a match, only joint balance can find it.

This example shows that if joint balance is of interest, it should always be directly checked; when it's not, it's still better than differences or marginals in that it maintains power and identification, with the caveats mentioned in section 4.3.8, i.e. when doubling the number of predictors is detrimental to prediction.

4.3.10 Prediction and Propensity

The conceptual framework behind judging matches by predictability is the same framework that justifies using the propensity score to build the matches in the first place. However,

[30] Or by direct proof using $g_m(x, y \mid 1, 0) = g_m(y, x \mid 1, 0)$.

maximising a function that depends on the whole match rather than just the distance matrices is difficult, hence checking predictability after is not the same problem as matching for predictability before, especially matching for pairwise absolute differences.

The connection between predictability and conditional independence is directly from the working definition of strong ignorability, given in Rosenbaum and Rubin's original work on the propensity score (Rosenbaum and Rubin 1983).

As noted by Pearl et al. (2009), this makes matching an excellent estimation technique, not a magic causal bullet.

One might ask, how do propensity matches not already maximise the prediction criteria laid out? Our post-match evaluation is different in a few ways to propensity scores. One way is that it doesn't involve units that aren't matched on. Units that are different enough to others in terms of $\mathbf{x}_i$ to not get matched on could easily bias our propensity score method in terms of its accuracy for units that are matchable. Secondly, it's a pairing method: we've already seen post-match densities for which a paired probability will not give answers similar to comparing two unpaired probabilities. [31] Further, as discussed in section 4.4.4, if the propensity match produces a lot of repeated control units, it will be a very predictable match.

4.4 Match Selection

Armed with our match metrics, we can now select a match from $\boldsymbol{\mathcal{M}}$, our set of matches.

4.4.1 Brier Selection

Our preferred metric for selecting matches is the Brier statistic, from section 4.3.6. Essentially, we evaluate $B(\boldsymbol{M})\ \forall \boldsymbol{M} \in \boldsymbol{\mathcal{M}}$, and select the match with the worst, i.e. highest[32]

[31] An example: it is not easy to predict the gender of a married working adult in the US as a function of their salary in the sense that you'd be wrong a lot. However, within a married couple ("paired") who both work, men earn more 82% of the time.

[32] Lower Brier scores indicate higher predictability.

Brier score, or when appropriate, the largest p-value.

Score over p value

We separate these two evaluations for two reasons. Firstly, if matching is difficult[33], we could easily have $\mathbf{p}_{\mathrm{B}}(\boldsymbol{M}) = 0$ for every match in our set. In such a case, we still want to do as well as possible, even if we won't achieve some version of "insignificance". Secondly, the permutation distribution is not actually of primary interest, and differs for every match: we want the best match, not the match least indistinguishable from a distribution based on the resulting covariates independent of treatment. A Brier score of 0.31 is worse than 0.29, even if the first produces a p value of 0.4 and the second a p-value of 0.6. Picking the largest p-value can select for highest variance of the resulting covariate distribution, unlikely to be a target of interest for us.

Secondly, from a performance point of view, computing a permutation Brier score with an expensive classifier is very expensive: even a noisy permutation estimate of just 100 permutation samples will cost 100 times as much as fitting a single match[34]. In many cases of matching with replacement, calculating the Brier scores is far more expensive than finding the matches, thus the entire process will become 100 times more expensive. This either results in extremely long computation time, or a very small number of matches. On top of this, permutation distributions are highly correlated between matches.

Following on from this, we usually only calculate the permutation distribution of the best match.

Supplementary Metrics

We have some mild exceptions: as mentioned in section 4.3.6, we compute a two-sided p value $\mathbf{p}_{\mathrm{B}}(\boldsymbol{M})$ because we want to avoid picking the most overfit dataset as our match: this

[33] For example, large imbalances between treated and control units

[34] Whatever we did to calculate the match's Brier should be used for each permutation sample. For example, if the match's Brier is evaluated with 5-fold cross validation, 100 permutation samples will require the model to be fit 500 times. This adds up even for e.g. logistic regression, not a method we advise.

could be as simple as wanting to avoid an unlucky distribution that happens to best fool our classification algorithm, and could result in unmeasured biases being amplified. The one-sided value $\mathbf{p}_{\mathrm{B}}^{\mathrm{L}}(\boldsymbol{M})$ is mainly to define the procedure.

Thus if many matches are above e.g. a median permutation score, or perhaps many are above the median permutation score of even just the best match selected, we could select the closest match to the median. This seems unprincipled.

Instead, we look at the implication of multiple matches all being very unpredictable: the Brier score judges these matches to be all very good in terms of verifying the reasonableness of 4.24, that the joint distribution is balanced. We then judge these acceptable matches by a less noisy metric: the Mahalanobis imbalance Δ_{Σ}[35] from section 4.3.3. In general, this is not our primary metric because it has less power under general joint distribution imbalance, but it does not suffer the same noise as our algorithm dependent Brier score does. Finally, while using weighted Mahalanobis imbalance is perfectly justifiable, changing the weights is less so: when comparing matches, even those generated by potentially different weight vectors, we should use exactly one version of Mahalanobis imbalance for comparison; this avoids any scale issues.

Permutation, not $1/4$

By the definition of the Brier score, if we know that exactly half the units are treated, a "default" guess would be to set $\hat{T} = 0.5$ for all units or pairs. This would give a Brier score of $1/4$, and thus provides a theoretical benchmark of signal in prediction.

However, if the data really has no signal, we would be lucky to learn this and set all predictions to $1/2$. For out of sample, or cross-validated, predictions, we will likely do worse than a null model; for in-sample, we will likely do better.

How badly a model will overfit a null model depends on what algorithm is used, and the structure of the data. Permutation analysis avoids having to understand these issues to

[35] If using weighted Mahalanobis, this would then become $\Delta_{\boldsymbol{w},\Sigma}$

generate a benchmark.

Distribution of Permutation Brier Scores

Figure 4.1 gives an example of the distribution of permutation Brier scores from our simulation study, (section 4.9), with $N = 500$. As the number of pairs used becomes small, the variance of the permutation distribution increases. Perhaps less expected, the mean decreases with the number of pairs.

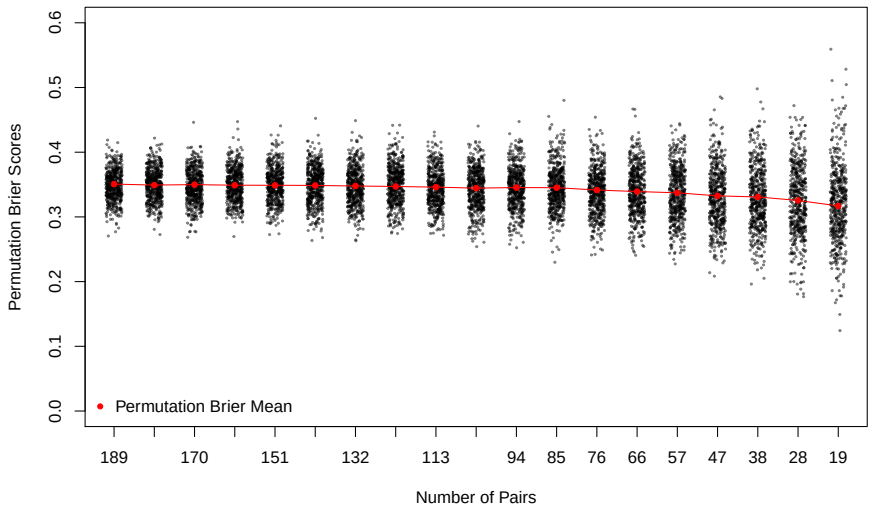


Figure 4.1: Sampled permutation Brier scores at different numbers of pairs. The mean of the distributioin decreases as the number of pairs decreases, while the variance increases.

This shows that we cannot use the same baseline permutation distribution if we change n, the number of pairs used in our match. This is in contrast to searching over matches for a given n, as we discuss in section 4.4.4.

4.4.2 Brier Selection Procedure

Following from the previous section, we have a full selection procedure, based on a suitable p value cutoff, p_{cut}. This method uses the Brier Testing Procedure from section 4.3.8. We assume we have some set of matches $\boldsymbol{\mathcal{M}}$ to select from.

Selecting from $\boldsymbol{\mathcal{M}}$

1. Evaluate the Brier statistic $B(\boldsymbol{M})$ for all $\boldsymbol{M} \in \boldsymbol{\mathcal{M}}$.

2. Let $\boldsymbol{M}_b$ be the best match, i.e. the match with the highest Brier score. Evaluate the full vector $\boldsymbol{B}_b^\star$ of permutation Brier scores, and $\mathbf{p}_{\mathrm{B}}(\boldsymbol{M}_b)$.

3. If $\mathbf{p}_{\mathrm{B}}(\boldsymbol{M}_b) \geq p_{\mathrm{cut}}$, stop. Return $\boldsymbol{M}_b$ as the selected match. That is, if the p-value indicates this match is reasonably within the distribution of permutation Brier scores.

4. Else: if we miss low, i.e. the one-sided p-value $\mathbf{p}_{\mathrm{B}}^{\mathrm{L}}(\boldsymbol{M}_b)$ is small, we also stop and return $\boldsymbol{M}_b$: this might not be ideal, but we've still selected the best of an imperfect group.

5. Else: we will compute pseudo-p values (two sided) for all matches by using $\boldsymbol{B}_b^\star$ from $\boldsymbol{M}_b$ instead of re-computing $\boldsymbol{B}^\star$ for every match[36]. Let $\boldsymbol{\mathcal{S}} \subseteq \boldsymbol{\mathcal{M}}$ be the set of matches with pseudo-p $\geq p_{\mathrm{cut}}$, i.e. all matches that are reasonable with respect to a permutation distribution[37].

6. If $\boldsymbol{\mathcal{S}} = \varnothing$, again return $\boldsymbol{M}_b$. Else calculate $\Delta_\Sigma(\boldsymbol{M})$ for all $\boldsymbol{M} \in \boldsymbol{\mathcal{S}}$, i.e. the Mahalanobis imbalance[38].

7. Return the match with the smallest Mahalanobis imbalance.

If the selected match is not very clearly better than the others, we may also report the results of the matches that are nearly indistinguishable from the selected match as an exploratory version of an uncertainty estimate.

The best match is a function of the set of matches (and the data). Let $\boldsymbol{M}_b$ be the best match, and we won't carry the full notation of $\boldsymbol{M}_b(\boldsymbol{\mathcal{M}}, \mathbf{X}, \mathbf{T})$ when we don't need to from context.

[36] This is mainly for computational purposes, and that in practice, permutation Brier scores don't vary a huge amount. This can be explicitly tested of course, and in section 4.4.4 we will discuss an issue with assuming the permutation scores don't vary.

[37] If $\boldsymbol{\mathcal{S}}$ is small, we can consider calculating the real permutation p values for each match.

[38] We pick one weight vector to compare Mahalanobis imbalance, we don't use the weight vector that generated the match

We'll discuss p_{cut} in the next section.

4.4.3 Evaluating the Best Match

Once we have our best match $\boldsymbol{M}_b$, we probably want to know if it's a good match, not just that it's the best of our set of matches. As alluded to in the selection procedure, one way of judging the match is its p-value, or equivalently, where its Brier score lands among the distribution of permutation Brier scores.

From standard statistical practice, $p_{\text{cut}} = 0.05$ seems reasonable, but the simulation study in section 4.9 implies higher values perform better.

This suffers the same issue as any classical statistical method does when selecting for performance: the performance estimate of the model with the best performance is biased.

Of course this refers to a form of external validity, and is not what we're trying to maximise. We want in-sample[39] balance, not out of sample balance. Based on this, we are happy to use the p-value of the best match to judge if it is indeed a good, balanced match.

Note of course there is nothing at all special about $p_{\text{cut}} = 0.05$, as we're doing essentially the opposite of hypothesis testing, hence why the simulation study does not cause us concern by deviating from 0.05.

4.4.4 Brier Scores: Matching With Replacement

Matching with replacement suffers from one prediction deficiency: a well-fit classifier can "learn" units that are used repeatedly as controls.

To be more specific, say control unit j is used as a control for treated units a and b. Say unit a and b end up in separate folds in our Brier prediction step. The classifier trained on the pair (a, i) will very likely correctly classify (b, i), and vice-versa. This is not an algorithm deficiency.

[39] Here, in-sample balance does not mean we use in-sample prediction, only that we're interested in the balance of the sample data we actually have.

Thus, matches with many repeated control units will naturally appear predictable. We view this as a feature, not a bug: all else being equal, repeating controls really does cause more distributional imbalance[40], and should be judged as such. This prevents potentially deficient matches being judged as well-balanced by simply repeating the small number of "good" controls many times, even when there is poor covariate overlap prior to matching.

If judging different matches on Brier scores, generally the permutation distributions will not be that different among matches; however large differences in the number of repeated controls between matches can slightly alter the permutation distribution. When computation time allows, one should calculate as many permutation distributions as possible. Note that this is when we're searching within a set of matches with the same number of pairs n. If n changes, we must change our permutation distribution, as detailed in section 4.4.4.

4.5 Bias and Variance of Matches

The previous two sections detail a very important aspect of matching: they should produce balanced sets. As discussed in section 4.3.10, this comes from the same justification as propensity matches do. When matching exactly on the propensity score, our estimate for the ATE or ATT will be unbiased.

Having close matches is important, beyond having balance, be it joint or otherwise. Abadie and Imbens (2006) derive bias and variance for matching in general, and Abadie and Imbens (2016) derive asymptotic properties of the estimated propensity score match, showing that in most cases it's more efficient than the true propensity score, assuming a generalised linear specification for the true model and the same model for estimation, with an in-sample MLE estimate of that model for the propensity score[41].

[40] Of course with the exception of exact matches, in which this prediction "bug" cannot occur.

[41] Whatever about the true model, using a generalised linear specification for the estimated propensity score with an in-sample fit is the most common form of propensity matching.

With a given match $\boldsymbol{M}$, the paired estimator of ATT is of the form:

$$\hat{\tau}_t = \frac{1}{N_t} \sum_{j=1}^{N_t} Y_{\boldsymbol{M}_{j,1}} - Y_{\boldsymbol{M}_{j,2}} \tag{4.53}$$

4.5.1 Bias

Let's focus on $\mathbf{E}[\hat{\tau}_t]$. We have $\mathbf{E}[\hat{\tau}_t] = \mathbf{E}[Y_u - Y_v]$, where u is a treated unit from our match and v is a control unit.

Theorem 4.5.1. *Assuming balance in the matching distribution, $f_m(x \mid 0) = f_m(x \mid 1)$, and further assuming that the matching distribution of treated units equals the population distribution, i.e. $f_m(x \mid 1) = f_1(x)$, the ATT from the match is unbiased.*

Proof.

$$\begin{aligned} \mathbf{E}[Y_u] =& \mathbf{E}[\mu(x,1)] \\ =& \int_x \mu(x,1) f_m(x \mid 1) \, \mathrm{d}x \\ \mathbf{E}[Y_v] =& \mathbf{E}[\mu(x,0)] \\ =& \int_x \mu(x,0) f_m(x \mid 0) \, \mathrm{d}x \end{aligned} \tag{4.54}$$

If we have distributional balance, the expected value of $\hat{\tau}_t$ can be written:

$$\begin{aligned} \int_x (\mu(x,1) - \mu(x,0)) f_m(x) \, \mathrm{d}x = \\ \int_x \tau(x) f_m(x) \, \mathrm{d}x \end{aligned} \tag{4.55}$$

Where $f_m(x) = f_m(x \mid t)$ for both $t \in \{0, 1\}$. If further we have $f_m(x \mid 1) = f_1(x)$, and thus $f_m(x) = f_1(x)$, then the above is the definition of τ_t. □

Since joint balance implies balance, we immediately get the following:

Corollary. *Assuming joint balance in the matching distribution, i.e. equation 4.24, and*

assuming that the matching distribution of treated units equals the population distribution, i.e. $f_m(x \mid 1) = f_1(x)$*, the ATT from the match is unbiased.*

Using the same methods, we also have an equivalent for the ATE:

Corollary. *Assuming balance in the matching distribution,* $f_m(x \mid 0) = f_m(x \mid 1)$*, and further assuming that this balanced matching distribution* $f_m(\mathbf{x})$ *is equal to the unconditional distribution of* $\mathbf{x}$*, i.e.* $f_m(x) = f(x)$*, the ATE from the match is unbiased.*

Note that having $f_m(x \mid 1) = f_1(\mathbf{x})$ is easy if we simply use all treated units in our match.

There are many cases where we don't require $f_m(x \mid 1)$ to be the same as $f_1(x)$ to get unbiasedness. For example, if a Feasible Average Treated Effect on the Treated is the target, then $f_m(x \mid 1)$ can define the target. If the treatment effect is constant, then we have unbiasedness with the above, or with difference symmetry, equation 4.29.

In general it's difficult to analyse the situations where we don't have balance, but the bias in that case will be smaller the closer the treated and control units are. Without any assumptions on $f_m(\mathbf{x}, 0)$, we can write:

$$
\begin{aligned}
&\mathbf{E}[Y_u - Y_v] \\
&= \int_x \mu(x,1) f_m(x \mid 1)\, \mathrm{d}x - \int_x \mu(x,0) f_m(x \mid 0)\, \mathrm{d}x \\
&= \int_x \Big(\mu(x,1) f_m(x \mid 1) - \mu(x,0) f_m(x \mid 1)\Big)\, \mathrm{d}x - \int_x \Big(\mu(x,0) f_m(x \mid 0) - \mu(x,0) f_m(x \mid 1)\Big)\, \mathrm{d}x \\
&= \int_x \Big(\mu(x,1) - \mu(x,0)\Big) f_m(x \mid 1)\, \mathrm{d}x - \int_x \mu(x,0) \Big(f_m(x \mid 0) - f_m(x \mid 1)\Big)\, \mathrm{d}x
\end{aligned}
\tag{4.56}
$$

The left term is τ_t if $f_m(\mathbf{x} \mid 1) = f_1(\mathbf{x})$, and is otherwise the FATT. The second term is the bias term. One can use e.g. Hölder's inequality (Hölder 1889) to get bounds in terms of $\int_x |f_m(x \mid 0) - f_m(x \mid 1)|^q \, \mathrm{d}x$ for various values of $q > 1$.

4.5.2 Variance

Let x be the covariates for unit u, and y the covariates for v. We'll assume the post-match densities are the same.

Variance: Paired Differences

If we have an unbiased method, then $\text{Var}(Y_u - Y_v) = \mathbf{E}\big[(Y_u - Y_v - \tau_t)^2\big]$. We can write:

$$\begin{gathered}(Y_u - Y_v - \tau_t)^2 = \\ (\mu(x,1) - \mu(y,0) - \tau_t + \varepsilon_u - \varepsilon_v)^2 = \\ (\mu(x,1) - \mu(y,0) - \tau_t)^2 + (\varepsilon_u - \varepsilon_v)^2 + \\ 2(\mu(x,1) - \mu(y,0) - \tau_t)(\varepsilon_u - \varepsilon_v)\end{gathered} \tag{4.57}$$

The last term has zero expected value and we can ignore it[42]. If we assume ε_u and ε_v are independent, the expected value of the right hand term $(\varepsilon_u - \varepsilon_v)^2$ will be $\mathbf{E}\sigma^2(x,1) + \mathbf{E}\sigma^2(y,0)$

The remaining term $(\mu(x,1) - \mu(y,0) - \tau_t)^2$ can be expanded in a similar fashion, to give

$$\begin{gathered}(\mu(x,1) - \mu(x,0) - \tau_t - \mu(y,0) + \mu(x,0))^2 = \\ (\mu(x,1) - \mu(x,0) - \tau_t)^2 + (\mu(x,0) - \mu(y,0))^2 + \\ 2(\mu(x,1) - \mu(x,0) - \tau_t)(\mu(x,0) - \mu(y,0))\end{gathered} \tag{4.58}$$

The last term again has zero expected value. We have $\mu(x,1) - \mu(x,0) - \tau_t = \tau(x) - \tau_t$, thus the term on the left is $\text{Var}(\tau(x))$, over the treated distribution. So we can write the

[42] Both terms have zero expected value: the first by unbiasedness, the second by construction of the residual terms.

full variance as:

$$\begin{gathered}\mathrm{Var}(Y_u - Y_v) = \\ \mathrm{Var}(\tau(x)) + \mathbf{E}\big(\mu(x,0) - \mu(y,0)\big)^2 + \mathbf{E}\sigma^2(x,1) + \mathbf{E}\sigma^2(y,0)\end{gathered} \tag{4.59}$$

Even assuming joint balance, the variance will be a function of the average differences in the mean of Y for the pairs. Hence we want our matches to be close, in order to make $\big(\mu(x,0) - \mu(y,0)\big)^2$ small. If we don't wish to assume that the errors are independent, our variance will be a function of violations of joint balance, in a manner that will be difficult to model.

Note that $\mathbf{E}\big(\mu(x,0) - \mu(y,0)\big)^2$ can be written as
$\mathrm{Var}(\mu(x,0)) + \mathrm{Var}(\mu(y,0)) - 2\mathbf{E}\left[(\mu(x,0) - \mu_0)(\mu(y,0) - \mu_0)\right]$, with $\mu_0 = \mathbf{E}[\mu(x,0)]$. The last term resembles covariance, and we want it to be large.

When we consider averaging over multiple pairs, assuming between-pair independence, we divide the variance by the number of pairs, n, to get the full variance:

$$\begin{aligned}\mathrm{Var}\left(\overline{Y_u - Y_v}\right) = \frac{1}{n}\Big(&\mathrm{Var}(\tau(x)) + \mathbf{E}\sigma^2(x,1) + \mathbf{E}\sigma^2(y,0) + \mathrm{Var}(\mu(x,0)) + \mathrm{Var}(\mu(y,0))\Big) \\ &- \frac{2}{n}\mathbf{E}\left[(\mu(x,0) - \mu_0)(\mu(y,0) - \mu_0)\right]\end{aligned} \tag{4.60}$$

Variance: Unpaired Differences

We detail unpaired differences, as most current analyses test post-matching samples as if they were independent. Note that the estimate is the same: paired vs unpaired matching has no effect on the bias, only the variance.

Schafer and Kang (2008) have said:

> "Matching" erroneously suggests that the resulting data should be analyzed as if they were matched pairs. The treated and untreated samples should be regarded

as independent, however, because there is no reason to believe that the outcomes of matched individuals are correlated in any way.

This would only be true if the mean functions were constant, in which case matching would not assist us[43]. It may be fully believable that the error terms are uncorrelated, or close to it, but this does not imply the actual outcomes are uncorrelated.

To see this, a typical correlated example of X and Y starts with some intermediate variable Z, with $X = Z + \varepsilon_x$ and $Y = Z + \varepsilon_y$, with $\varepsilon_x, \varepsilon_y$ independent noise. This gives correlated X and Y. If Z was known as a predictor, X and Y would still be correlated, even with independent errors. Matching on Z would give X and Y units that were closer in outcome than randomly paired X and Y.

Hill and Reiter (2006) find undercoverage of the matched pairs estimator, but they don't appear to correct for matching with replacement, as per section 4.5.4. Another point to make: if the methods aren't fully unbiased, and they likely won't be, nominal coverage will fall short as a function of the bias. Any variance increase, justified or not[44], will increase nominal coverage; we should not use this to conclude that this potentially unjustified increase therefore is better. Coverage is most important at treatment effects of zero, and bias tends to be strongest away from zero.

Using essentially the same techniques as in the previous section, we can show:

$$\mathrm{Var}\left(\overline{Y}_u - \overline{Y}_v\right) = \frac{1}{n}\Big(\mathrm{Var}(\tau(x)) + \mathbf{E}\sigma^2(x,1) + \mathbf{E}\sigma^2(y,0) + \mathrm{Var}(\mu(x,0)) + \mathrm{Var}(\mu(y,0))\Big) \tag{4.61}$$

Contrasting this to 4.60, the difference is $-\frac{2}{n}\mathbf{E}\left[(\mu(x,0)-\mu_0)(\mu(y,0)-\mu_0)\right]$. Therefore the unpaired variance is smaller if the pairs are closer in terms of their mean function than completely random pairings.

It is difficult to defend a matching scheme on its merits while also concluding that the

[43] With the exception of matching to find overlapping sets

[44] For example, simply doubling the variance

matched pairs it creates are not on average closer together than random pairings. That does not mean that in all circumstances researchers would want to analyse matches on the pair level, but it would seem the default choice should be to use pair matching, and only treat the samples as independent if there are other concerns besides variance.

4.5.3 Small Matching Distances

We use Mahalanobis matches as a baseline for matching distance. For a given weight vector $\boldsymbol{w}$, we can calculate the total Mahalanobis distance for any match, $D^{\boldsymbol{w},\Sigma}(\boldsymbol{M})$[45]. We also have the best match according to this metric, the Mahalanobis match $\boldsymbol{M}^{\boldsymbol{w},\Sigma}$, which minimises this metric by construction.

We define the ratio[46] of a match as the ratio of the total Mahalanobis distance for that match relative to the best possible Mahalanobis distance:

$$\mathrm{R}_{\boldsymbol{w},\Sigma}(\boldsymbol{M}) := \frac{D^{\boldsymbol{w},\Sigma}(\boldsymbol{M})}{D^{\boldsymbol{w},\Sigma}(\boldsymbol{M}^{\boldsymbol{w},\Sigma})} \tag{4.62}$$

Σ is dropped from the ratio notation when it can be assumed to be the same covariance matrix across the matches, and we'll just write $\mathrm{R}_{\boldsymbol{w}}$.

By definition, the ratio for a Mahalanobis match judged by the weight vector that created it will be 1, and all other matches will have $\mathrm{R}_{\boldsymbol{w}} \geq 1$, with equality only if the match is identical, or there are ties, rendering the Mahalanobis match non-unique.

For a caliper match, $\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e})$, we are most interested in its ratio $\mathrm{R}_{\boldsymbol{w}}$ for the same weight vector used to create it - that is, we want to compare caliper matches to the Mahalanobis matches that use the same weight vector. Let R define this:

$$\mathrm{R}(\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e})) := \mathrm{R}_{\boldsymbol{w}}(\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e})) \tag{4.63}$$

[45] Using the definition from equation 4.3

[46] Potentially weighted, i.e. a function of a chosen weight vector

Unfortunately this ratio R is not well-defined, even though $\mathrm{R}_{\boldsymbol{w}}$ is[47], when two weight vectors produce the same match, i.e. $\boldsymbol{M}(\delta_1, \lambda_1, \boldsymbol{w}_1, \Sigma, \mathbf{e}) = \boldsymbol{M}(\delta_2, \lambda_2, \boldsymbol{w}_2, \Sigma, \mathbf{e})$[48], but $\boldsymbol{w}_1 \neq \boldsymbol{w}_2$. This happens most frequently when considering the propensity match $\boldsymbol{M}^{\mathbf{e}}$: lemmas 4.2.1 and 4.2.2 show that for some large enough λ_1 and λ_2 and with $\delta_1 = \delta_2 = 0$, both of the above matches will equal $\boldsymbol{M}^{\mathbf{e}}$.

We can solve this by defining R as the minimum in ambiguous cases, i.e.:

$$\mathrm{R}(\boldsymbol{M}) := \min_{\delta,\lambda,\boldsymbol{w}} \left\{ \mathrm{R}_{\boldsymbol{w}}(\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e})) : \ \boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e}) = \boldsymbol{M} \right\} \tag{4.64}$$

Based on this, we can use a cutoff R_{cut} such that we ignore all matches with $\mathrm{R}(\boldsymbol{M}) > \mathrm{R}_{\text{cut}}$. This provides us in general with something like a bias-variance trade-off - in some cases, this might not be a trade-off at all if the variance is also decreasing with closer matches.

4.5.4 Matching With Replacement: Variance Correction

When we match optimally, the variance estimate is straight forward. We can use:

$$\hat{V}_1 = \frac{1}{N_t} \frac{\sum_{i=1}^{N_t} \left(Y_{\boldsymbol{M}_{i,1}} - Y_{\boldsymbol{M}_{i,1}} - \hat{\tau}_t \right)^2}{N_t - 1} \tag{4.65}$$

This is the familiar variance of differences: we'd get this by typing
`var(Y[M[,1]] - Y[M[,2]]) / nrow(M)` into the language R, assuming `M` is the match matrix as we've been using it, and `Y` is the outcome vector.

When we match with replacement, we have to account for the dependence between control terms. Abadie and Imbens (2006) propose a consistent estimator for the asymptotic variance of $\hat{\tau}_t$.

Let C_j be the set of control units matched to treated unit j, being empty if unit j is a

[47] Of course assuming matching covariance matrices
[48] We could also have $\mathbf{e}_1$ and $\mathbf{e}_2$ potentially different, although we usually don't.

control unit, i.e. $C_j = \varnothing$ if $T_j = 0$. Define:

$$K_i = \begin{cases} 0 & \text{if } T_i = 1 \\ \sum_{i=1}^{N} \frac{\mathbf{1}(i \in C_j)}{\mid C_j \mid} & \text{if } T_i = 0 \end{cases} \tag{4.66}$$

K_i is the weighted number of times unit i is used as a control. For most of our matches we do nearest neighbor matching, so each control set is empty or of size one, i.e. $\mid C_j \mid \in \{0, 1\}$, which means K_i is the number of times a unit is used in our match.

Similarly, define $K_{\text{sq},i}$:

$$K_{\text{sq},i} = \begin{cases} 0 & \text{if } T_i = 1 \\ \sum_{i=1}^{N} \frac{\mathbf{1}(i \in C_j)}{\mid C_j \mid^2} & \text{if } T_i = 0 \end{cases} \tag{4.67}$$

In nearest neighbor matching, we have $K_{\text{sq},i} = K_i$.

We would like to get:

$$V_2 = \frac{1}{N_t} \sum_{i=1}^{N} (K_i^2 - K_{\text{sq},i}) \sigma^2(\mathbf{x}_i, T_i) \tag{4.68}$$

Note that the summand will always be zero if $T_i = 1$, so really we might as well write $\sigma^2(\mathbf{x}_i, 0)$. Of course we don't know $\sigma^2(\mathbf{x}_i, T_i)$ so we have to estimate it. The proposed method uses nearest neighbour matching for each control unit with another control unit[49] to estimate the variance. Let $l(i, j)$ be the jth closest control unit to control unit i; $l(i, 1)$ is the closest control unit to control unit i (excluding i). We use:

$$\hat{\sigma}^2(\mathbf{x}_i, 0) = \frac{J}{J+1} \left(Y_i - \frac{\sum_{j=1}^{J} Y_{l(i,j)}}{J} \right)^2 \tag{4.69}$$

Typically we'll just use the closest value, i.e. $J = 1$. Note that non-bipartite matching must adjust this formula.

[49] And if we needed $\hat{\sigma}^2(\mathbf{x}_i, 1)$, we match treated to treated.

This gives us $\hat{V}_2$. Finally, $\hat{V}_1 + \hat{V}_2$ is the variance we use.

4.6 Generating Matches

Now that we have a procedure to select a match given a set of matches, we would like to simply generate all matches, as in section 4.3.1, and run the procedure. This is completely impossible even in small data situations.

Instead, we will generate a subset of all possible matches, with an eye to generating matches that will perform well in our procedure, while also sufficiently exploring the space of all matches.

4.6.1 All Caliper Matches

Equation 4.12 defines a caliper distance for a given caliper δ and multiplicative factor λ. And as usual, we can replace Mahalanobis distance d_Σ with weighted Mahalanobis distance $d_{\boldsymbol{w},\Sigma}$ in our definition of caliper distance.

Let $\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e})$ be the caliper match formed from using caliper width δ, multiplicative factor λ, weight vector $\boldsymbol{w}$ in the Mahalanobis match along with Σ, and $\mathbf{e}$ the propensity score in the propensity match; we'll continue to use $\boldsymbol{M}^{\delta,\lambda}$ as the caliper match when it's clear from context we're holding the Mahalanobis and propensity distances fixed.

We can see by inspection that $\delta = 1$ or $\lambda = 0$ both result in (weighted) Mahalanobis distance, i.e. $\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e}) = \boldsymbol{M}^{\boldsymbol{w},\Sigma}$, and lemmas 4.2.1 and 4.2.2 tell us that there is some large value of λ such that caliper distance matching gives us propensity matching, i.e. $\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e}) = \boldsymbol{M}^{\mathbf{e}}$.

Indeed, every value of $(\delta, \lambda) \in [0, 1] \times [0, \infty)$[50] defines a caliper distance, and this set of distances gives us a finite[51] set of caliper matches that cover the spectrum from Mahalanobis

[50] At actual infinity, the caliper match won't recover propensity matching because the distances will be equal outside of the caliper.

[51] The set of "all" caliper matches must be finite, because it's a subset of all possible matches for a given dataset, which is finite.

matching up to propensity matching.

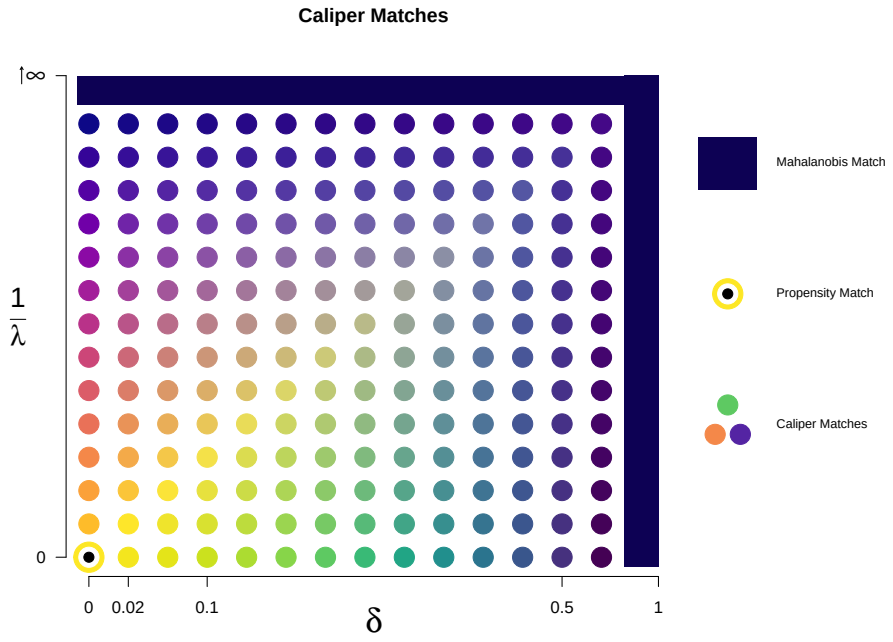


Figure 4.2: We compute caliper matches at a grid of values. Unsurprisingly, a linear search is not the most effective. The entire block of matches at $\delta = 1$ or $\lambda = 0$ give the same base Mahalanobis match. Lighter colours indicate proximity to the propensity match.

Figure 4.2 provides a conceptual diagram, and alludes to our plan: we search along a grid of values for δ and λ. The different colours are to convey that the various paths one can take from the Mahalanobis match to the propensity match can produce quite different caliper matches.

For an example, with $\delta = 0.1$ and $\lambda = 10^9$, our match will generally agree with Mahalanobis matching, but where it doesn't, it could be very far. For matching with replacement, this means many pairs will remain the same, but the pairs that change might end up matching units that seem far away. In contrast, $\delta = 0.005$ and $\lambda = 5$ gently encourages propensity matches' favoured controls, and could result in nearly every pair changing, although they won't change to pairs that Mahanalobis distance strongly discourages.

Grid Search For Calipers

Trying "all" values of (δ, λ) is impossible, and extremely similar values will result in essentially the same match, so is also pointless. Further, we don't wish to actually find every single match, because the testing procedure would become computationally infeasible, let alone the feasibility of storing the matches.

Instead, we find a reasonable search space, ideally to explore as much of the caliper space between the two end-point matches as possible for a given number of matches.

For δ: Mahalanobis matches already try to bring together pairs with similar covariates. Similar covariates generally implies similar propensity scores, thus the propensity distances in Mahalanobis matches are not in general terrible. For us, this means it's usually pointless to set δ to be anything more than about 0.2 times the range of $\mathbf{e}$. This isn't equivalent to the usual recommendation of setting $\delta = 0.2 \times sd(\mathbf{e})$, mainly because we're not looking for a single best caliper, only an upper limit[52]. We will always go down to zero, to create the linear interpolation calipers, i.e. so that the caliper distance is a linear combination of the Mahalanobis and propensity distance.

For λ: first we find a value that creates propensity matches for $\delta = 0$, and set this as our maximum. We don't quite need to go to $\lambda = 0$, because we just get the Mahalanobis match, but we should go close.

The actual values depend on the Mahalanobis scale: if we have e.g. $\sum \boldsymbol{w} = 1$, then in our urban data, $\lambda = 0.01,\ 0.05,\ 1,\ 10$ and $500,000$ all provide a good exploration of the space.

Further, setting $\boldsymbol{\delta} = (0,\ 0.02,\ 0.05,\ 0.08,\ 0.12,\ 1) \times R$ where R is the range of the propensity score, $\max \mathbf{e} - \min \mathbf{e}$, provides a good balance of calipers in our urban matching.

Good grid values for $\boldsymbol{\delta}$ will depend on the grid used for λ and vice versa. In practice, one should explore various values for both, and evaluate how well the space is explored. Metrics

[52] In the most extreme case, we'll have $sd(\mathbf{e}) = 0.5$; typically it will be much lower.

to look at include the evaluation criteria, i.e. the Brier score and the Mahalanobis imbalance, the total Mahalanobis distance, and the number of pairs that are the same as either of the two end-point matches[53]. One can also simulate data, and use that to inform the grid.

Looking again at Figure 4.2, we note that the diagonal is always further from propensity matches than the horizontal and vertical lines leading away from the diagonal, towards the x and y axes. This explains the shape of some of our simulation figures. The reason is because for any caliper along a line from the diagonal, e.g. with δ_1, λ_1 as the parameters, the "diagonal" caliper match δ_2, λ_2 either has $\delta_2 = \delta_1$ and $\lambda_2 < \lambda_1$, or $\delta_2 > \delta_1$ and $\lambda_2 = \lambda_1$.

Simulation Example

We show part of our full simulation study detailed in section 4.9; here we use propensity model C, the sign model, and outcome model E, the cubic model. We have five covariates, $N_t = 413$ treated units and $N_c = 487$ control units.

We use a highly compressed $\boldsymbol{\delta}$ of length 11, with many values near zero[54]; for $\boldsymbol{\lambda}$, we again use eleven, but much larger values[55]. In theory this gives 121 matches, but 21 of them are just the Mahalanobis match[56], so these are just one match, for a total of 101; one of these is the propensity match, thus at most 99 new matches.

We first test our grid. Figure 4.3 plots the total Mahalanobis distance, i.e. $D^{\boldsymbol{w},\Sigma}(\boldsymbol{M})$, for every match, divided by the total Mahalanobis distance of the Mahalanobis match. Since the Mahalanobis match by construction minimises this difference, the ratio starts at one and builds towards the propensity match's ratio.

In terms of finding an appropriate grid, we use this plot along with counting the number of pairs, or potentially counting the number of controls, that are the same in each caliper match

[53] This is a better metric for matching with replacement, since optimal matching tends to shuffle controls around even for small changes in the distance.

[54] We use $\boldsymbol{\delta}$ = (0, 0.0005, 0.001, 0.0029, 0.0096, 0.019, 0.039, 0.068, 0.097, 0.19, 0.96), rounding to two significant digits; the maximum difference between propensity scores here is 0.96, hence it's the largest value.

[55] We use $\boldsymbol{\lambda}$ = (5,000,000, 10,000, 5,000, 2,000, 500, 100, 10, 3, 1, 0.1, 0).

[56] All eleven with $\delta = 1 \times R$, and all eleven with $\lambda = 0$, minus the double counted ($\delta = R, \lambda = 0$).

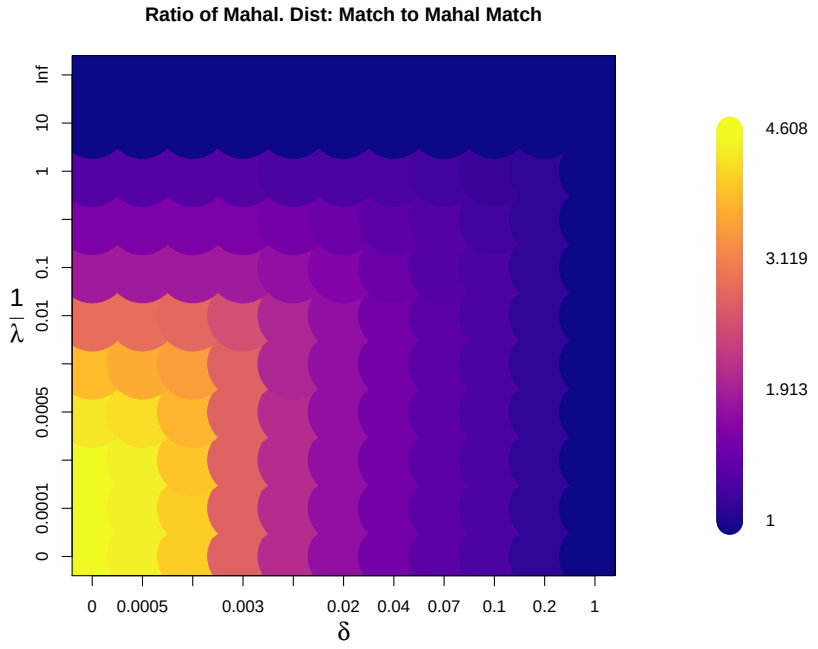


Figure 4.3: For each caliper match at each point of our grid, we calculate the total Mahalanobis distance for that match, and divide by the total Mahalanobis distance for the base match. The bottom left is the propensity match. The x-axis is actually δ times the range of the propensity scores.

relative to the propensity and Mahalanobis matches. Poorly chosen grids will generate many matches too close to one and not close enough to the other.

Figure 4.4 plots the Brier score of each caliper match against its estimate for the treatment effect. Here, we simulated a true effect of 1. As is typical, the caliper matches bounce around the space near both the Mahalanobis match and the propensity match. The green line represents the caliper matches if one follows the diagonal path in Figure 4.2: starting at the largest δ and smallest λ, increase both incrementally along the grid of values until we get to the smallest δ and largest λ[57].

In this particular example, the best caliper was chosen from the maximum λ value, and a

[57] This "diagonal" only exists if the number of values of each parameter is the same, $| \boldsymbol{\delta} | = | \boldsymbol{\lambda} |$. We don't require a square grid in applied work, and the diagonal is not of any particular importance, it's just a simple graphical outcome to represent.

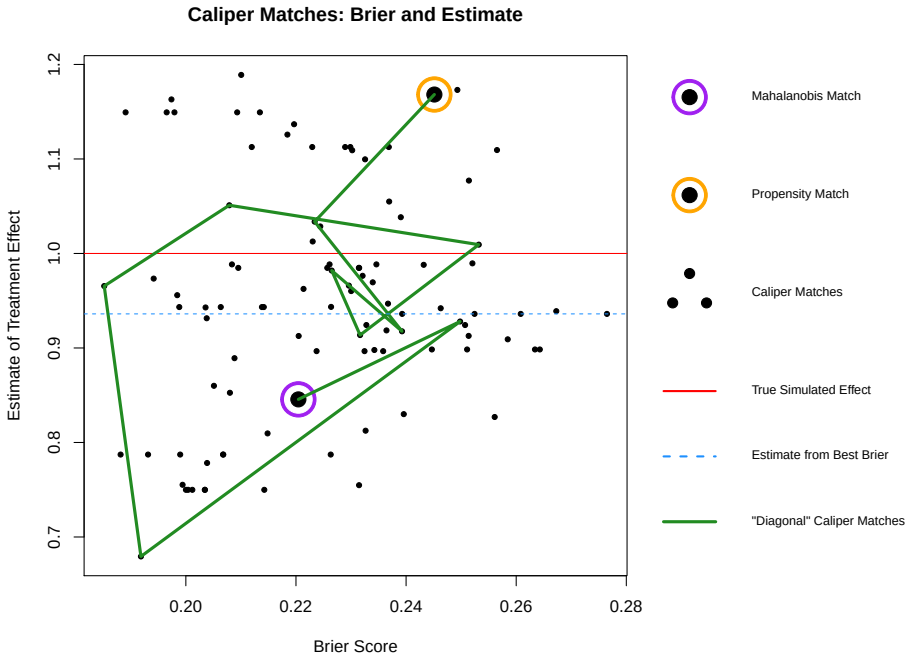


Figure 4.4: We compute caliper matches at a grid of values; we calculate the Brier score for each match, and then plot Brier scores against the estimate of the treatment effect from that match. The green line is one path from the Mahalanobis match to the propensity match.

middling value of δ[58], but produced almost the same match as the next closest, with the same δ value and λ near the middle of the grid, $\lambda = 500$.

Calipers in Practice

We note that the above example uses a larger grid than we use in our applied work. Mainly this is due to the fact that searching over calipers is generally less effective than searching over weights, which we detail in the next section. Since every base Mahalanobis match produces a grid of caliper matches, it can become very expensive very quickly to run huge grids over huge numbers of base matches.

Further, as noted in section 4.5, we want our matches to be close. Thus we often apply a ratio restriction, as detailed in section 4.5.3. In our simulations, $\mathrm{R}_{\mathrm{cut}} = 2$ performed well[59],

[58] $\delta = 0.02 \times R$, or $0.08 \times sd(\mathbf{e})$ if preferred.

[59] The best value for squared error was generally lower while the best value for bias was higher.

so that's what we use. It would be prudent to continue to monitor matches that violate the constraint, for example if the propensity match achieves excellent balance and none of the allowed matches do, the variance tradeoff is unlikely to be worthwhile.

4.6.2 All Weight Vectors

Before now, we've mentioned that Mahalanobis matches can use a weighted option instead of treating all variables equally, with or without using ranks.

Similar to searching over the space of caliper matches, we will also search over the space of weight vectors.

The Space of Weight Vectors

In section 4.1.2, we discuss that distances are only identified up to a multiplicative constant: using d or $14 \times d$ will produce the same minimising solution, thus the same match. We scale our weight vectors to always add to one. Of course, all elements must be positive, thus we can define the set $\boldsymbol{\mathcal{W}}$ of all weight vectors uniquely:

$$\boldsymbol{\mathcal{W}} := \{\mathbf{w} \in \mathbb{R}_+^p : \sum \mathbf{w} = 1\} \tag{4.70}$$

Where p is the number of covariates, and $\mathbb{R}_+$ is the set of positive reals[60].

This space is huge, we don't attempt to search all of it.

Biased Random Searching

One could search over grids of values, similar to caliper searching. Other options include iteratively searching through weights based on the results of previous searches. The GenMatch algorithm in the R packaging Matching (Sekhon 2011) performs such a search.

An alternative is to randomly search over the simplex $\boldsymbol{\mathcal{W}}$. The most typical implementation

[60] We mean ≥ 0, not > 0, although we often force $\mathbf{w}$ to be non-zero in every element.

of such a search is a Dirichlet distribution. It's easy to intentionally bias such a search, and reasonable when one has prior beliefs that certain variables are more important than others. This doesn't mean we always force high weights to variables we think should get high weights, only that we bias the search space to increase the likelihood of high weights.

One can also search with avoidance to maximise the search: if computational time allows for e.g. $1,000$ matches to be searched[61], one could generate far more than $1,000$ weight vectors and subsample.

Similarly, we may wish to force a lower bound on some or all of our weights. This is easily implemented in a random search.

Search Space for Matches

Regardless of what method we choose, we end up with a set of weights $\boldsymbol{\mathcal{V}} \subset \boldsymbol{\mathcal{W}}$, which we will use to form our set of matches $\boldsymbol{\mathcal{M}}$.

Simulation Example

We show part of our full simulation study detailed in section 4.9; here we use propensity model C, the sign model, and outcome model D, the non-linear model. We have five covariates, $N_t = 364$ treated units and $N_c = 436$ control units. We generate 300 weight vectors uniformly over the simplex.

Figure 4.5 plots the Brier score against the output. While not very interesting in simulation, in applied cases we can examine the weight vectors that produced the best matches to see what variables the match procedure deemed most important. Similar to the caliper case, we plot the best match according to the Brier score.

This example was chosen to show how dramatic weight searching can be. The treatment probability only depends on one variable, but the outcome does not; the Brier method selects

[61] This refers to the entire matching procedure. Generating random weight vectors is very cheap computationally relative to the rest of the matching procedure.

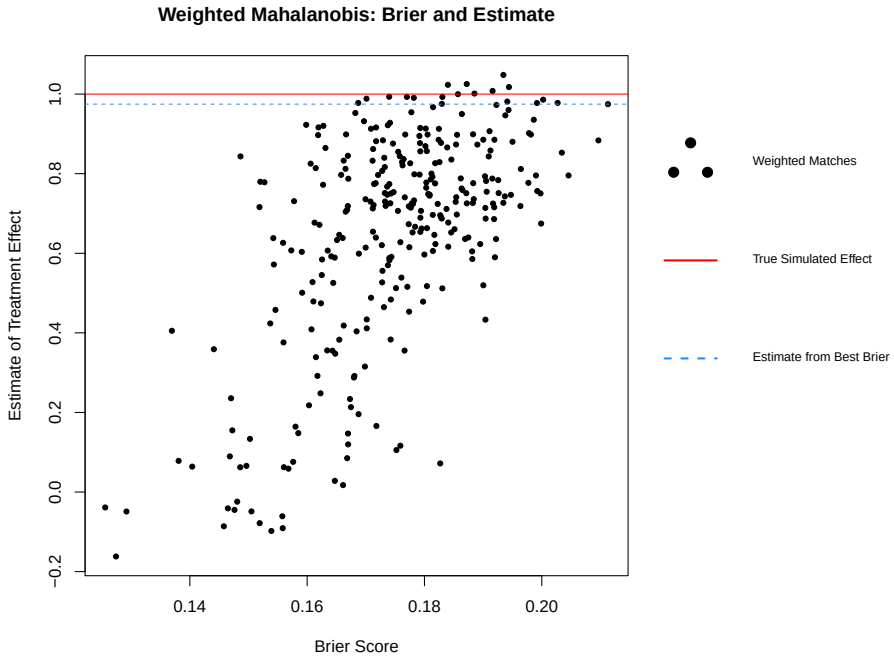


Figure 4.5: We compute many Mahalanobis matches, with different weight vectors. The red line is the true simulated treatment effect, and the blue line is our outcome estimate.

a weight vector with a large weight on this variable, but not the largest, since ignoring the other variables can still cause the match to make imperfect pairs.

Further, in this example, a linear model produces a poor fit of 0.58 with a standard error of 0.09, and the simple estimate of using all treated and all controls to form an unpaired estimate is negative.

4.6.3 Multiple Propensity Matches

Once we've chosen our algorithm, e.g. boosting or logistic regression; our distance function, e.g. 4.9 or 4.10; and if we want in-sample or out-of-sample propensity scores, we can define a propensity match. But we don't have to stop there: we could for example generate multiple propensity matches using multiple algorithms. This can lead to even more caliper matches: a set of matches over the grid for each propensity match.

In our work, we do not find a huge benefit to such a search. For example, logistic propensity

matches perform far worse than boosting, unless the functional form of the treatment probability suits logistic. This mirrors the findings of Lee, Lessler, and Stuart (2010)[62]. Thus we use just one propensity match.

4.6.4 Generating Our Full Set of Matches

Now that we can create caliper matches for every weight vector, and we have a set of weight vectors, we can create a large set of matches.

First, we set our caliper grid vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\delta}$[63], and our ratio cut-off $\mathrm{R}_{\mathrm{cut}}$. The process:

Generating $\boldsymbol{\mathcal{M}}$

1. Initialise $\boldsymbol{\mathcal{M}} \leftarrow \varnothing$
2. Generate a propensity score $\mathbf{e}$, and find the covariance matrix Σ, rank-adjusting if necessary.
3. Generate a set of weight vectors $\boldsymbol{\mathcal{V}}$
4. For every weight vector $\mathbf{w} \in \boldsymbol{\mathcal{V}}$, generate the caliper matches for every (δ, λ) pair in the grid. For every match generated, $\boldsymbol{M}(\delta, \lambda, \boldsymbol{w}, \Sigma, \mathbf{e})$, calculate the Mahalanobis ratio $\mathrm{R}(\boldsymbol{M})$. If $\mathrm{R}(\boldsymbol{M}) \leq \mathrm{R}_{\mathrm{cut}}$, add it to the set of matches, if not already there:

$$\boldsymbol{\mathcal{M}} \leftarrow \boldsymbol{\mathcal{M}} \cup \{\boldsymbol{M}\} \tag{4.71}$$

Combining this with the Brier selection procedure outlined in section 4.4.2 establishes our search methodology.

[62] That is, boosting out-of-sample beats both in-sample and out-of-sample logistic. In-sample boosting requires a lot of fine-tuning.

[63] Really, we run through part of the process and evaluate the matches generated to see if we need to change $\boldsymbol{\lambda}$ and $\boldsymbol{\delta}$, until they produce caliper matches that span a reasonable space.

4.7 Optimal Number of Matches

We don't have to use all units: there is no guarantee that we can find a good match for all our treated units. The variance increase due to throwing away units can be easily offset by the bias reduction of avoiding terrible matches. In fact in many cases, the variance is also reduced, due to closer matches being formed. This is also a consequence of section 4.5.2.

The main downside is we lose the general interpretation of the ATT, and instead estimate a Feasible Average Treatment Effect on the Treated, FATT. We say "a" not "the", because the number of matches used in fact defines the FATT: as we reduce the number, we are likely to move the post-match density of the treated $f_m(x \mid 1)$ and the controls $f_m(\mathbf{x} \mid 0)$ closer together. Whatever distribution they converge to defines the FATT.

Let N_m be the number of matches we are happy with, with $N_\text{m} \leq N_t$. Let's call the optimal match for a given distance function d with N_m pairs $\boldsymbol{M}(N_\text{m})$, a function of N_m. As in section 4.1.2, we order the treated units to define the match uniquely up to ties.

$$\boldsymbol{M}(N_\text{m}) := \underset{\boldsymbol{M}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N_\text{m}} D_{\boldsymbol{M}_i} : \ \boldsymbol{M}_{k,1} < \boldsymbol{M}_{l,1} \text{ for } k < l \right\} \tag{4.72}$$

Note that this does not give the same results as keeping the best N_m matches from $\boldsymbol{M}(N_t)$[64].

King, Nielsen, et al. (2011) plot matches against n, the number of units matched. They are generally comparing a Mahalanobis match, a propensity match, and a coarsened exact match at each number of matches, with various balance metrics evaluated. They conclude that one should choose the best of the three matches for whatever value of n is selected. We extend this by using a more general metric to compare large sets of matches, and guidance for choosing a stopping value for n if the researcher hasn't decided in advance.

[64] Quick example: assume we have to match football teams, and we have 50 college teams, one highschool team and one NFL team. The optimal match in terms of skill will likely force the best college team to match with the NFL team, and the worst with the HS team, although those would be terrible matches. If we were allowed to drop two matches, those would certainly be dropped first. However, if we only needed to form 24 matches instead of the full 26, naturally the NFL and HS team would not get matched, along with two other college teams. The best and second best college team could be a great match, and would now be a possible pair.

4.7.1 Choosing the Optimal Number of Matches

We need criteria for choosing N_{m}, the number of matched pairs in our final match. The larger, the more matches; the smaller, the better the matches.

Distance Cutoff

Rosenbaum (2012) constructs pareto-optimal optimal[65] matches according to distance cutoffs. He recommends typical cutoffs based on the full distance matrix D, such as the twentieth percentile. Let d_{20} be this value. Then we choose N_{m} to be the largest n such that $\boldsymbol{M}(n)$ creates a match with all distances less than this cutoff. In our notation:

$$N_{\mathrm{m}} = \max\{n : \max_i D_{\boldsymbol{M}(n)_i} \leq d_{20}\} \tag{4.73}$$

Deciding what is "close enough" will depend on prior knowledge. Another candidate is to form non-bipartite matches with all our units: matches that ignore the treatment condition, but still minimise the total distance between the pairs. These non-bipartite matches solve equations that are analogous to equations 4.2 and 4.72, except we don't have any restriction on $T_{\boldsymbol{M}_{i,1}^{\mathrm{nbp}}}$ or $T_{\boldsymbol{M}_{i,2}^{\mathrm{nbp}}}$, and the upper limit in the sum in equation 4.2 is $\lfloor N/2 \rfloor$. We discuss non-bipartite matches further in section 4.8. After these matches are formed, we set our cutoff based on the worst match formed; we can also allow less than $\lfloor N/2 \rfloor$ in the NBP comparison, e.g. we could use $\lfloor 0.9 \times N/2 \rfloor$, essentially throwing away the worst 10%.

Permutation Brier Cutoff

As already discussed in section 4.4.3, we can use the permutation Brier distribution, or equivalently the Brier p-values, to judge a match: once the match is sufficiently unpredictable, we are satisfied and can stop. The idea behind this being a useful metric is the same idea behind judging matches using the Brier testing procedure.

[65] Not a typo - optimal means two different things here.

Mahalanobis Imbalance

We can also keep reducing n until we get sufficiently small Mahalanobis imbalance, Δ_Σ, as in section 4.3.3.

Of course, as Ho et al. (2007) point out, there is no "good enough" balance - we should keep trying to improve balance as much as possible[66]. As we'll discuss, this is not necessarily concerning.

4.7.2 Metrics for every n

The above metrics can potentially stochastically improve all the way until we set $n = 1$, which is useless. Since we choose a match before looking at the outcome, we aren't barred from trying a whole bunch of values of n, and monitoring the metrics at every stage. In our simulations, it's common to see practically no improvement below some number of matches. This offers reasonable lower bounds on n.

To give an example: say we reduce n from $N_t = 1000$ down all to way to $n = 100$ in a given problem. The Brier scores could stochastically improve until about $n = 600$, at which point they might seem to more or less bounce around 0.32 as n heads all the way to 100. After checking for e.g. Mahalanobis imbalance for these matches with $n \approx 600$ and judging them sufficient, we would not likely use $N_\text{m} < 600$. Why would we? We'd be throwing away units for no real reason, and moving the FATT further from the ATT.

Note that propensity matching suffers a unique issue: King and Nielsen (2016) show that as we send $n \to 0$, we don't get uniform improvement in propensity matches on a collection of metrics.

Our main method is to reduce n until the Brier score for our selected match is sufficiently inside the permutation Brier distribution.

[66] That's not the same as improving Brier scores as much as possible - once we're overfitting, it's hard to justify higher scores meaning better balance.

4.7.3 Full Matching Procedure

We now outline our full matching procedure, using the Brier selection procedure in section 4.4.2 (which uses the Brier testing procedure in section 4.3.8), and the match generating procedure in section 4.6.4. We need to set everything those algorithms require.

Full Match Procedure

1. Set $n = N_t$
2. Using the match generation procedure, generate a large set of matches with a large set of weight vectors, by generating a set of caliper matches for each weight vector.
3. Evaluate this set using the Brier selection procedure. Let $\boldsymbol{M}_{\text{best}}(n)$ be the resulting match.
4. If the Brier p-value of this match, $\mathbf{p}_{\text{B}}(\boldsymbol{M}_{\text{best}}(n))$, and the Mahalanobis imbalance $\Delta_{\Sigma}(\boldsymbol{M}_{\text{best}}(n))$, satisfy pre-set requirements, stop. $\boldsymbol{M}_{\text{best}}(n)$ is our selected match.
5. Else $n \leftarrow n - 1$. Return to step 2.

We could set $p_{\text{cut}} = 0.05$, Due to the selection bias issue, a higher cutoff requirement is probably better. Again, the simulation study in section 4.9 implies higher values, e.g $p_{\text{cut}} \geq 0.5$, perform better. We don't use an explicit cut-off for Δ_{Σ}.

In practice, the largest jump is from having no matches with a p-value other than zero for large n to at least one match having a non-zero p-value, at least in large matches. Once a single match gets close to the permutation distribution, it generally won't take much further reduction in n to reach reasonable limits.

Practical Issues

In practice, we generate matches on a grid of possible values for n, from N_t down to e.g. $\lfloor 0.1 \times N_t \rfloor$. We compute the best match for value of n, and look at the metrics for each.

In fact, similar to Figures 4.4 and 4.5, we can plot the Brier scores and any other metrics, such as Δ_Σ, to see the general behaviour of our matches. Note that unlike those figures, we obviously cannot plot the outcome if we haven't yet chosen a match.

Note that Brier scores are usually stochastic: it's worth investigating the variance of Brier scores for fixed matches, as we don't want to select a match based on noise.

We also don't generally regenerate the set of weight vectors, i.e. in step 2. We generate a set once, and use those for every n. This helps in two ways: it provides a more reasonable comparison between different values of n. We detail the second benefit below.

Matching With Replacement and n

Generating a single match allowing replacement isn't that expensive, but computing the distance matrix can be, and each weight vector generates a different distance matrix.

When matching with replacement for a given n, it is the case that removing the worst match gives the best match for $n-1$, and so on.

Once we've got a set of weights $\boldsymbol{\mathcal{V}}$, and our grid of calipers, we can in fact generate a whole bunch of matches for different values of n, and store those to run the procedure.

Treatment Effect as $n \to 0$

Without wanting to adjust significant values Bonferroni style (Bonferroni 1936), we can't look at the outcome of multiple matches. However, for exploratory analysis, we might not care about significance. Further, we can report our main result, i.e. our result for the match we select along with the n we select, and then see what other matches would have reported.

One way to view outcomes is to look at the best match for each n, from N_t down to any small number, generally smaller than our selected number N_m. As the number of matches decreases, the bias shrinks, the variance may grow or shrink, and sadly the target moves from the ATT to the FATT. However, if the FATT is of interest, this plot can be very useful,

and we'll see examples in the results section of the next chapter, section 5.3.2.

4.8 Non-bipartite Matches

We've already discussed non-bipartite matches, or NBP matches, in section 1.1.4. We'll discuss adjusments in our algorithms required to use them in NBP matches.

As mentioned in section 1.1.4, the assumptions from 1.1.1 change slightly for NBP matching. Overlap becomes $\varepsilon < \mathbf{P}(T = t \mid X) < 1 - \varepsilon \ \forall \ t, X$ for some $\varepsilon > 0$; ignorability becomes $Y_t \perp\!\!\!\perp T \,\middle|\, X$. The concept of ATT might not make as much sense anymore. SUTVA remains unchanged.

4.8.1 NBP Distance

The concept of distance between units is very similar to bipartite matching. In fact, all versions of Mahalanobis remain exactly the same. For propensity, instead of predicting zeros and ones, we just predict the treatment, on whatever scale it is. For example, if treatment is a continuous dosage, we try to predict the dosage.

The difference between NBP and bipartite matching is not our main focus. We are mainly interested in the treatment being univariate and continuous, and will assume that structure going forward.

4.8.2 NBP Treated and Control Units

NBP has no clear treated and control units. In the case of continuous treatment, we'll assume the direction of the treament is meaningful: in a matched pair, the unit with the larger value of treatment is the treated unit, and the other is the control.

We've assumed up until this point that the match is represented by a matrix $\boldsymbol{M}$, where the first column is an index of treated units and the second an index of control units. This doesn't work for NBP matching.

Instead, we'll make a small adjustment: for a given row in $\boldsymbol{M}$, the first element's value of T is larger than the second element[67], i.e. $T_{\boldsymbol{M}_{i,1}} > T_{\boldsymbol{M}_{i,2}}$. This does not mean that every element of the first column has a larger treatment value than every element of the second, since for example we could pair units with treatment values 100 and 90 together, and also pair units with 60 and 50; the "control" unit in the first match has more treatment than the treated unit in the second.

Minimum Separation

Implicit in this definition is that a match can't be formed between units with the same treatment value. This is reasonable, since we don't learn about the effect of the treatment in such a pair. To take this notion further, we often apply a restriction on the treatment difference in pairs, i.e. some minimum value they must be seperated by in order to be matched. Let's call this value $\mathrm{t}_{\min}$.

The point of such a separation is usually an identifiability worry, or similarly a variance issue: we want to learn how differences in the treatment value affect the outcome, so if we fail to separate the treatment values, the differences in outcomes will be mainly driven by bias if there is any, and will be swamped by variance.

To take a simple case, assume a linear treatment effect, $\mu(\mathbf{x}, t) = \mu(x) + \beta t$. Our expected difference in matching Y_i with Y_j:

$$\begin{aligned} \mathbf{E}[Y_i - Y_j] &= \mathbf{E}[\mu(x_i, T_i) - \mu(x_j, T_j)] \\ &\quad \mathbf{E}[\mu(x_i) + \beta T_i - \mu(x_j) - \beta T_j] \\ &\quad \mathbf{E}[\mu(x_i) - \mu(x_j)] + \beta(T_i - T_j) \end{aligned} \tag{4.74}$$

Similarly, one can look at the variance, e.g. from section 4.5.2. If the match is unbiased, then the expected difference is $\beta(T_i - T_j)$. But if the difference is very small, e.g. $T_i - T_j = \nu$

[67] This is true in our bipartite definition too; we can even use this defition for both

for some small[68] ν, then the effect could be nearly unidentifiable; or even worse, even a very slight bias in our match could mean we're just measuring our match's bias.

Say we want to look at scaled effects, what we called the "derivative" of the effect $\tau(\mathbf{x}, \delta)$ in section 1.1.4, i.e looking at terms of the form:

$$\frac{Y_i - Y_j}{T_i - T_j} \tag{4.75}$$

We would need a reasonable minimum to avoid variance blowups as the denominator approaches zero.

We'll say a match $\boldsymbol{M}$ is $\mathrm{t}_{\min}$-separated if all treatment values in the pairs differ by more than $\mathrm{t}_{\min}$, i.e.:

$$\min_i \left\{ T_{\boldsymbol{M}_{i,1}} - T_{\boldsymbol{M}_{i,2}} \right\} > \mathrm{t}_{\min} \tag{4.76}$$

We use strict inequality so that $\mathrm{t}_{\min} = 0$ is well-defined. $\mathrm{t}_{\min} = 0$ is common if for example T only takes values in a finite set with suitable separation.

An NBP match always has a minimum separation, even if it's zero, or negative in the rare cases we wish to allow potentially equal matches[69]. We can write $\boldsymbol{M}(\mathrm{t}_{\min} = \eta)$ to mean the match $\boldsymbol{M}$ is η-separated.

Maximum Separation

We can also place a maximum, $\mathrm{t}_{\max}$, on the differences in matched treatments.

The main reasons we would do this are for overlap and ignorability: we might not believe both hold for all T for a given X, but we might be more confident in local effects.

If we believe the effect is locally linear, then we don't want to match hugely different values

[68] Small in absolute value, relative to the variance.

[69] For example, the cross match test (Rosenbaum 2005) and related tests, or using unconstrained matches to check how much of a restriction the treatment restriction is, e.g. Rosenbaum (2012), e.g. in section 4.7.1.

of the treatment together. We will discuss this further in the next chapter, chapter 5, when we match on opening hours.

We may have two types of businesses[70], e.g. one type that is typically open most of the day, and another that is not. We might be interested in the effect of an extra hour or two each day on crime, believing the effect of the extra hour could be similar for both types of businesses. For example, a restaurant open from 8am to midnight, compared to another that opens from 9am to 11pm. They will differ by two hours a day. In contrast, a dinner service only restaurant might operate from 6pm to 10pm, and a longer one 5pm to 10.30pm. The difference here is 90 minutes a day. Such pairs might combine linearly to answer the question: what are 1.45 extra hours worth? However, if we instead crossed up the pairs, we might no longer believe the comparison can be made the same way: at some point there is some normalisation, so a twelve hour difference might look very different than six times a two hour difference.

Finally, imposing maximum separation can serve to decrease the number of matches with differences close to the minimum in certain circumstances, while simply increasing the minimum might have knock on effects we don't want.

Similar to minimum separation, $\mathrm{t}_{\max}$-separation means:

$$\max_i \left\{ T_{\boldsymbol{M}_{i,1}} - T_{\boldsymbol{M}_{i,2}} \right\} \leq \mathrm{t}_{\max} \tag{4.77}$$

We write $\boldsymbol{M}(\mathrm{t}_{\max} = \omega)$ to mean the match $\boldsymbol{M}$ is ω-separated.

4.8.3 Minimising Distance

NBP matching has a less obvious target to minimise than bipartite matching. We still want to have something that looks like $\sum_i D_{\boldsymbol{M}_i}$.

First we can define the optimal matches, where every unit can only be used once. Without

[70] Even within the same business category

any further restriction, let $N_{\text{half}} = \lfloor N/2 \rfloor$, i.e. half the number of pairs, rounded down if N is odd.

We essentially have the same equation as equation 4.4: the best match minimises the sum of distances. Of course unlike in that bipartite case, we allow a unit to be either a treated or control, so we have different implicit assumptions on the match matrix $\boldsymbol{M}$, including potentially $t_{\min}$- and $t_{\max}$-separation.

Matching with replacement is a little trickier: we more or less maintain equation 4.2, and we likely still want our set of treated units to be unique, but we may or may not want to block units from showing up as both treated and control.

4.9 Simulation Study

We test our method with a varied simulation study. The first purpose of the study is to verify that our method produces answers that are better than current matching methods. Secondly, we wish to compare against a regression benchmark. Thirdly, while we try to avoid fixing parameters where possible, we have introduced a permutation p-cut off p_{cut} and a ratio cut-off R_{cut}, thus we analyse the best values for both. Since we don't assume we have the right model, we are not directly interested in coverage probabilities.

We generate a factorial design: five different models for $\mathbf{P}(T = 1 \mid X)$, i.e. true treatment probability models, and five models for $\mu(X, 0)$. To provide a strong benchmark, we simplify the joint outcome such that $\mu(X, 1) = \tau + \mu(X, 0)$. In other words, we model $\mu(X)$, and let $\mu(X, T) = \mu(X) + \tau T$. This benefits regression solutions, even with a non-linear function $\mu(X)$.

Given a vector of covariate $\mathbf{x}$, our five probability models:

A None: the probability of treatment is the same for all units. We fix this at 0.425.

B Linear: a logistic fit, i.e. we generate β such that $\mathbf{P}(T = 1 \mid X) = \text{expit}(\beta' X)$. The intercept is set so that we generally get 40% treated. We generated $\beta \sim \mathcal{N}(0, 0.2^2)$

C Sign: the probability of treatment is determined by the sign of X_1, the first element of the vector. If $\text{sgn}(X_1) = 1$, the probability is 0.7, else 0.2.

D Non-linear: we use $\text{expit}\big(\text{sgn}(X_1 - 1)/2 + \beta' X\big)$, also with $\beta \sim \mathcal{N}(0, 0.2^2)$.

E Cubic: we use $\text{expit}\big((X_2^3 - 2X_2^2)/10\big)$.

For $\mu(X)$, we have a similar set:

A None: this sets $\mu(X)$ to zero - since we're computing differences, the overall mean isn't relevant.

B Linear: we generate β such that $\mu(X) = \beta' X$. This is not the same β as in the treatment model; we generated $\beta \sim \mathcal{N}(0, 0.2^2)$

C Sign: the mean is determined sign of X_1, but we switch it half the time: thus for the even iterations we have $\mu(X) = \text{sgn}(X_1)$; for the odd we have $\mu(X) = -\text{sgn}(X_1)$

D Non-linear: we use $\mu(X) = \cos(X_1) + \sin(\beta' X) \times \text{sgn}(X_3) - \tan(\pi/3 - \min_j\{|\ X_j\ |\} \times \pi)$, with $\beta \sim \mathcal{N}(0, 0.04^2)$.

E Cubic: we use $\mu(X) = \text{sgn}(X_2) \times \left(\sqrt{2\overline{X^2}} + X_3^3\right)$, where $\overline{X^2}$ means the mean of X^2.

The number of total units is either 500, 1,000, 2,000 or 4,000. The number of covariates is 5, 10, 20, or 40. X is generated according to a correlated normal distribution with random correlations and all variances equal to one, or an independent uniform distribution for each variable on $(-1, 1)$.

We set $\tau = 1$, and generate $Y_i = \mu(\mathbf{x}_i) + \tau T_i + \varepsilon$. To make our regression benchmark even stronger, we let $\varepsilon \sim \mathcal{N}(0, 1)$.

For every simulation we generate either 50, 100, 150 or 300 weight vectors, and the same 26 caliper matches as we do for our applied problem[71]. Finally, we generate 19 different numbers of pairs, starting from the number of treated units, going down to 10% of the

[71] $\boldsymbol{\delta} = (0,\ 0.02,\ 0.05,\ 0.08,\ 0.12,\ 1) \times$ the range; $\boldsymbol{\lambda} = 500,000,\ 10,\ 1,\ 0.05,\ 0.01,\ 0$. Eleven of these give the same Mahalanobis match.

number of treated units, by 5% each time.

4.9.1 Results

The basic metric for evaluation will be squared error, with bias a secondary metric. For each simulation, we use our matching procedure from section 4.7.3 to select a match, for different values of p_{cut} and R_{cut}.

p_{cut} ranges from zero to 0.975 in increments of 0.025, giving a grid of length forty. Recall that this is a two-sided p-value, thus a value close to one implies the match is close to the center of the permutation distribution. When the restriction is zero, we will always use the maximum number of pairs, since there is no p-value requirement.

R_{cut} is based on quantiles of ratios, and ends up with many values near zero, stretching to two and beyond[72].

Note that a larger p_{cut} value is more restrictive, since it's a minimum value we require our matches meet; a larger R_{cut} is more permissive, since it's a maximum value our matches must stay under.

Squared Error

For every pair of values $(p_{\text{cut}}, \text{R}_{\text{cut}})$, we can compute $(\hat{\tau} - 1)^2$ for all estimates $\hat{\tau}$. This is not just variance: squared error is variance plus the bias squared.

We can then average over all simulations, and all twenty five model pairings, to get the average squared error value for each pair.

Figure 4.6 plots the results of this analysis for the smallest matches: $N = 500$, with N being the number of total units. There is not much difference in our results between different numbers of covariates for X, so we just average over those. There are a few things to note. Importantly, the optimal p-values are large, from about 0.325 to 0.6. In fact, using a small

[72] Most ratios are very close to the Mahalanobis ratio of 1. The full vector we test is (1, 1.0003, 1.001, 1.01, 1.03, 1.12, 1.25, 1.3, 1.35, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2, 2.4, 3, 4, 10, ∞).

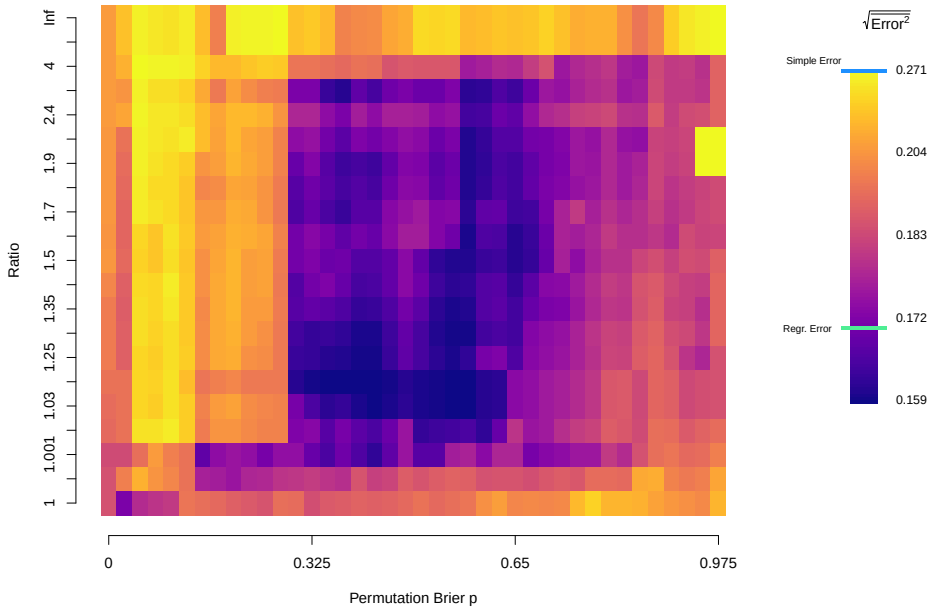


Figure 4.6: We plot the square root of the average squared error, for each pair of inputs to our procedure: the p-value we cut off at, and the ratio we require matches to be under. Note that the legend contains the respective regression and simple estimates. This plot averages over all simulations with $N = 500$.

p-value cut off is worse than no cut off: this is because, at least for small N, the tradeoff from lowering the number of pairs in using any non-zero cutoff is large, and is only overcome at a reasonably large p-value.

Using $p_{\text{cut}} = 0.5$ would mean that the p-value must be in the middle 50% of the permutation distribution. As the requirment becomes more strict, the matches do worse because there are fewer to check Mahalanobis imbalance on.

The best ratio is somewhat small, the optimal $\text{R}_{\text{cut}} = 1.12$. However, the penalty for increasing this retriction is small up to essentially only ruling out pure propensity matches.

The regression estimate is good in comparison to our distribution of errors, but is worse than the optimal pair, and many pairs near the optimal pair. The regression error is not plotted directly, but is included in the legend of the plot. This error becomes significantly

worse as we increase N: linear regression becomes more susceptible to extrapolations, and our matches become better with sample size.

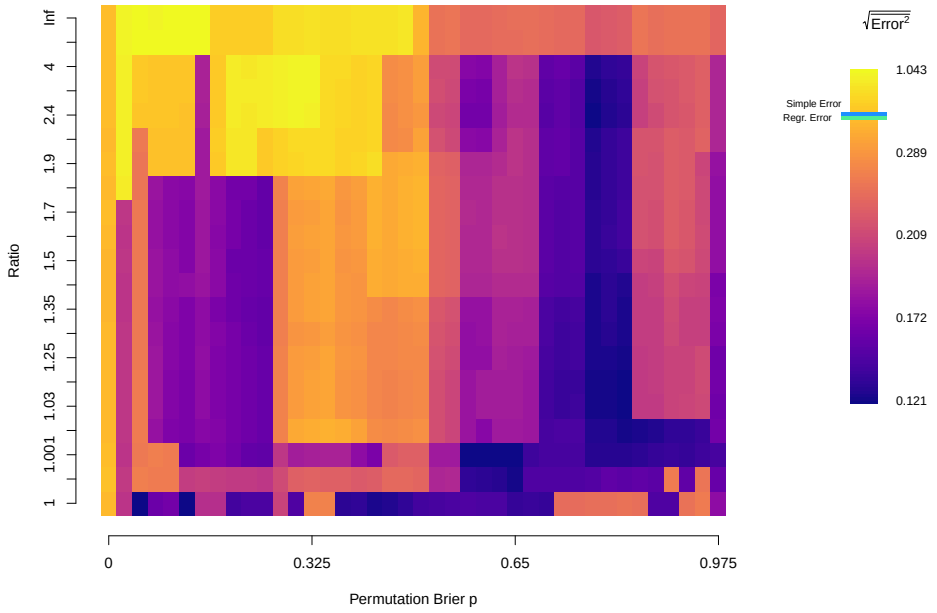


Figure 4.7: We plot the square root of the average squared error, for each pair of inputs to our procedure: the p-value we cut off at, and the ratio we require matches to be under. Note that the legend contains the respective regression and simple estimates. This plot averages over all simulations with $N = 1000$.

Figure 4.7 is the same plot, but now for $N = 1000$. As discussed, regression is now worse than most of these pairs: only the worst combinations of p_{cut} and R_{cut} are worse than our procedure. Note the scale is not linear.

The simple error, i.e. unmatched pair differences, also improves with sample size. With $N = 500$, the simple error is worse than all procedure pairs. With $N = 1000$, it's about as bad as linear regression.

Results for $N = 2000$ and $N = 4000$ resemble $N = 1000$: any sensible pair beats linear regression, and of course beats the simple estimate.

Bias

It's not easy to fully separate bias and variance in the total error estimate, but we can try. For each pair of values $(p_{\text{cut}}, \text{R}_{\text{cut}})$, we average through the simulations individually for each of our twenty five match types. After averaging, we will assume we have a reasonable estimate of the bias for that model setup. We then can average squared bias across all models.

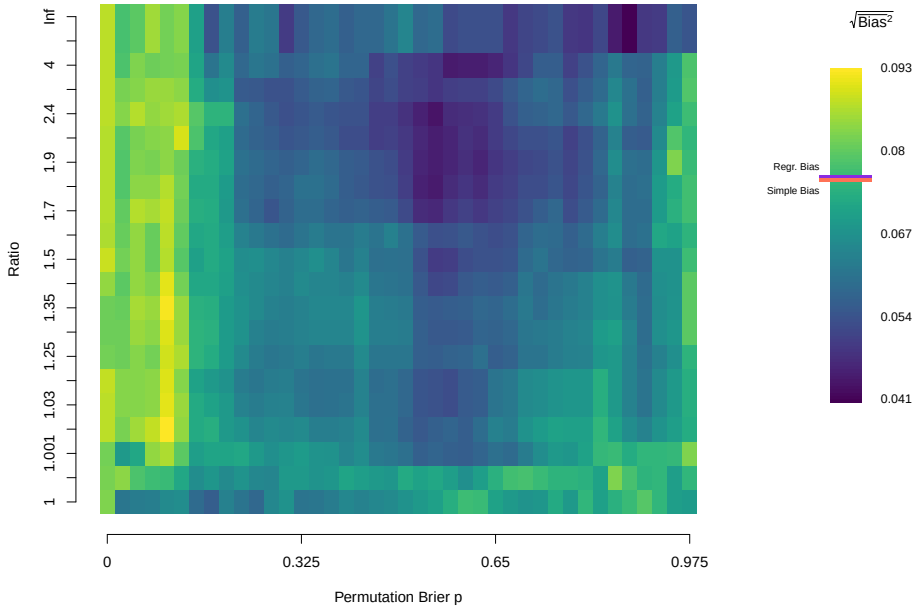


Figure 4.8: We plot the estimated bias, for each pair of inputs to our procedure: the p-value we cut off at, and the ratio we require matches to be under. Note that the legend contains the respective regression and simple estimates. This plot averages over all simulations with $N = 500$.

Like with squared error, the smallest case is the most interesting. We plot bias per pair of cutoffs in Figure 4.8. Note that the simple estimate is now better than the regression estimate, even though it's much worse in squared error.

Mainly, we'll note that the lowest bias results have large ratio cut offs. This makes sense, since that ratio is used mostly to control variance. The optimal p-values are also slightly

larger than when looking at squared error.

Bias makes up part of the squared error plot, hence why these aren't separate pieces of evidence. If error is your main concern, this plot is close to irrelevant.

On the other hand: variance is estimable from the sample. Thus we may be willing to sacrifice having slightly larger variance in order to reduce biases, which are harder to see from a given sample.

Regression Comparison

Let's focus on $N = 500$, where our method is only slightly better than regression. We might want to know when it's better and when it's not, in terms of the functional forms for treatment probability and outcome functions.

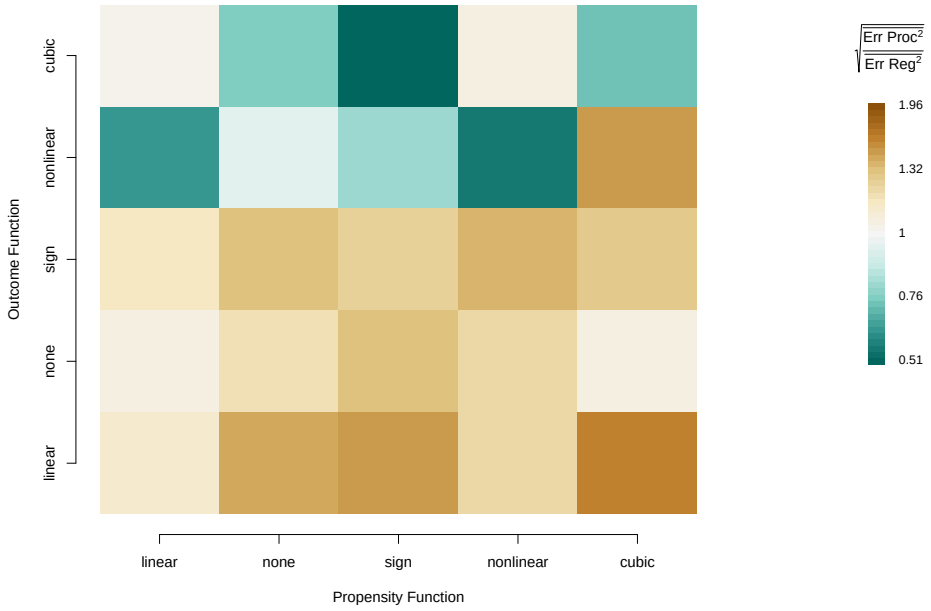


Figure 4.9: We pick one example of a matching procedure input, and plot the outcome error against regression error. Err Proc means the error from our procedure, and Err Reg is the error from using regression to estimate the treatment value. We use $N = 500$ here.

Figure 4.9 plots the error average of one pair of values $(p_{\text{cut}}, \text{R}_{\text{cut}})$[73], divided by the regression average error, then we take the square root. Note that the brown in the image is weaker than the green, altohugh it may not be easy to tell from the legend[74]. Our method beats regression because regression does really poorly when the outcome model is the non-linear model or the cubic model, so poorly that it loses overall.

It is no surprise to see regression perform well when the outcome function is linear since there is no more efficent model. When the outcome model is empty, regression is nearly most efficient, only losing to emptier models, which would be unknowable ahead of time. The sign model doesn't bias regression much, because extrapolation in such a case is minimal; although here our method is nearly as efficient. When the outcome models are truly wrong relative to assumptions, linear regression can produce some very poor results, even though the effect of treatment is fully linear and the error distribution is correctly specified.

For larger N, linear regression performs worse overall relative to the matching procedure, but the pattern here remains consistent: regression is worse because it performs extremely poorly in difficult cases. Of course we don't know the functional form of the outcome or the probability function in real work, but from a minimax point of view, linear regression is significantly worse. In our simulations, the maximum squared error of regression is eighty times larger than the maximum error for the matching procedure with $N = 500$[75], going up to about two hundred times worse for $N = 4000$. That is, regression has very poor worst-case behaviour.

Mahalanobis and Propensity Comparison

Our method must be an improvement over unselected matches to be worth the extra work. We'll compare our method to plain propensity matching and Mahalanobis matching with uniform weights.

[73] One of the better pairs - it has a lower squared error than regression.

[74] Since this is a ratio, 1 means equality, but 0.5 and 1.5 are not equally far away: 0.5 and 2 are.

[75] At well chosen permutation and ratio cut-offs. For typical values, it's about fifty times, and the worst possible case still has the procedure with a maximum error fifteen times smaller than regression's error.

For a fair comparison, we will also search over permutation p value cut offs for both Mahalanobis and propensity.

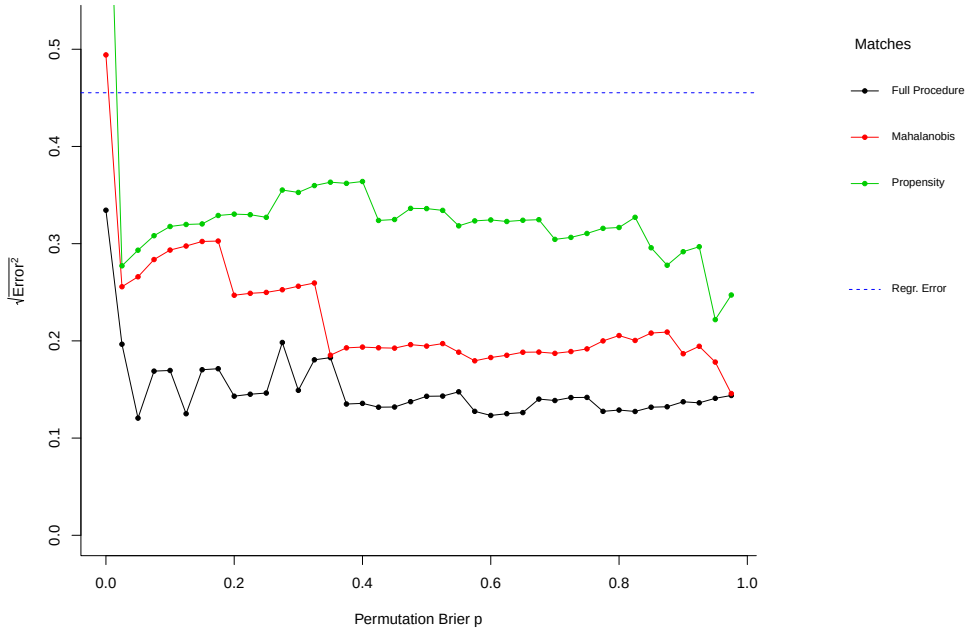


Figure 4.10: We plot the error rates at different p cut off values, comparing our full procedure to Mahalanobis and propensity matching. We use $N = 1000$ here.

Figure 4.10 plots the error rates for different permutation p cut offs, and includes the overall regression rate for comparison. This is for simulations with $N = 1000$. We'll note that in this case, the full procedure often picked Mahalanobis matches, thus the reason this specific Mahalanobis match is getting beaten is because the full procedure chooses the best of a set of Mahalanobis matches at every p_{cut} value, i.e. the best weighted Mahalanobis.

Propensity matching improves a lot by going from $p_{\text{cut}} = 0$, i.e. we take the propensity match with all pairs, to $p_{\text{cut}} > 0$: reducing the pair count until we hit the permutation distribution. But after, it gets worse until p_{cut} is close to one. Combining this with King and Nielsen (2016) implies that pure propensity matching should be used with caution.

The traditional comparison would be the full Mahalanobis and propensity matches: using all the pairs. This corresponds to $p_{\text{cut}} = 0$. We see that the Mahalanobis match error at

$p_{\text{cut}} = 0$ is poor, worse on average than regression with $N = 1000$, and the propensity error averege is very poor, 0.815, too large to fit on the plot.

4.9.2 Discussion of Simulation Results

Overall, we've seen that our full procedure on average is an improvement over typical matching benchmarks, Mahalanobis matching and propensity matching, and a very large improvement if those benchmarks use all the pairs.

Further, our method has significantly better worst-case error rates than linear regression, not to mention unmatched estimates.

Unfortunately the results do not unambiguously inform us what ratio and permutation p cut off values we should use: the best combinations depend on the number of treated and control units. As discussed in section 4.7.3 however, once we reduce the number of pairs used down to get any non-zero permutation p value, it generally doesn't take long to reach the centre of the permutation distribution. Thus we recommend using large values of p_{cut}: $p_{\text{cut}} \geq 0.5$ where possible. The desired ratio seems less vital, and gives a more direct bias variance trade-off. For this reason, we recommend a high cut off[76] of $\text{R}_{\text{cut}} \geq 2$.

4.10 Conclusion

In this section, we outline four procedures that combine to form our method:

1. The Brier Testing Procedure (BTP) in section 4.3.8, which tests a given match in terms of predictability

2. The Brier Selection Procedure (BSP) in section 4.4.2, which uses the BTP along with Mahalanobis imbalance to select a match from a set of matches

3. Our match generation procedure in section 4.6.4, which generates a large set of caliper matches from a set of weight vectors, bearing total distance in mind

[76] Unlike for p_{cut}, a higher R_{cut} is less restrictive, and allows more matches.

4. Our Full Match Procedure (FMP), which generates matches and selects a match for every n, helping the user to decide when to stop reducing the match size

4.10.1 Subsetting Our Procedures

It is not vital that users combine the entire procedure. If a researcher has decided on a particular weight vector and a number of matches n, but is unsure of which caliper to select, she can create a large set of matches and test them all. If a researcher has exactly one matching method in mind but hasn't decided on n, she can still use part of the procedure to learn a stopping point. Similar partial use is still helpful for randomising multiple weight vectors.

4.10.2 Pairwise For Power

We also recommend testing pairwise predictability rather than unpaired, or just differences. Pairwise covers all situations, with the best of both worlds in terms of power and identifiability.

As noted in section 4.3.8, typically exceptions to this are when the parameter space is very large, i.e. training and predicting on the smaller p dimensional scale is better than $2p$.

4.10.3 Future Work

When deciding between matches that are all low in Brier score, i.e. predictable, selecting the least predictable match is probably the safest move, as bias is likely the greatest concern.

As predictability decreases, it's hard in general to know the best bias variance tradeoff for any given problem, and while the BTP generates our preffered metrics, it's clear in simulation and in theory that when many matches are deemed good, maximising the Brier score without bound while ignoring Mahalanobis imbalance and the overall distance between the pairs is not a solid plan. We would like more concrete results in this area.

Chapter 5

Urban Analysis: Intersections

5.1 Matching

We now have what we need to perform matching. We use the matching methods detailed in the previous chapter, chapter 4: we search extensively through the space of possible matches until we satisfy our required criteria. To be specific, we use the Full Matching Procedure outlined in section 4.7.3, with the modifications from the following sections, sections 4.7.3 and 4.7.3: we search over a subset of the number of possible matches, from the maximum down to 10% of the possible maximum, and we use a fixed set of weight vectors for each n.

5.1.1 Matching Matrix and Setup

We combine all the data from chapter 2, specifically as it's detailed in section 2.3, where we detail how our intersections are created, and how each data type is involved on the level of intersections.

To recap, we have:

- Eight demographic variables, the five proportions of race (black, white, Asian, hispanic, other), total population, mean income, and our poverty metric. We also have one missing indicator for counts and proportions, and one for the two economic variables.
- Eight property variables (number of properties, average value, age and age deviation, average # stories, average # garages, price per square foot, price per square foot deviation), with three missing indicators: one for all property data missing, one for age variables, and one for square foot price variables.
- Three transit variables: number of bus routes, number of trolley routes, and number

of subway routes.

- Six school variables, three for high school and three for elementary school: distance to nearest, enrollment of nearest, and smoothed count density.
- Two traffic signal variables: an indicator for traffic lights, and an indicator for an all-way stop sign.
- For all matches except those in which a function of total number of businesses is the treatment, we match on the total number of businesses.

This gives twenty eight variables, with five extra missing indicators[1].

We draw gamma six distributed variables with weights 10, 5, 2, 3, 1, 4 respectively for the categories above, i.e. we give a lot of weight on average to demographics, and very little to traffic control. We divide these weights by the sum to scale them. Within each category, we split the assigned category weight up randomly by generating uniform random numbers and scaling by the sum[2], with minor exceptions: with demographics, we weigh total population, income and poverty all slightly higher than the five proportion counts; within property, we weigh total number of properties slightly higher. Finally, all non-missing indicators are forced to have a minimum weight of 0.01.

We use the caliper grid discussed in section 4.6: $\boldsymbol{\delta} = (0,\ 0.02,\ 0.05,\ 0.08,\ 0.12,\ 1) \times R$, and $\boldsymbol{\lambda} = (0,\ 0.01,\ 0.05,\ 1,\ 10,\ 500,000)$.

Every variable is rank-adjusted, and all matching is done with replacement, and with one control (nearest neighbour matching).

5.1.2 Matching Experiments Performed

We perform three main types of experments. Each has crime as the outcome. As noted in section 2.3.2, we split crime into two types, violent and non-violent, and we add the two to

[1] We tend to give them a small weight: nearly all restrictions that we make, e.g. to intersections with at least one business, rules out all nearly all intersections with missing data.

[2] This intentionally produces weights closer together than e.g. a uniform search over the simplex.

form total crime[3]. We also form two time sections, before and after business data collection: we run our analyses twice as a method of control, in the sense that different violations of assumptions can be thought to affect crime collected pre-data collection, and crime collected post-data collection. We will discuss this further in section 5.3.3.

We have 28 matches:

Presence Presence vs Absence of businesses: what is the effect of the presence of each of our business types on crime in the intersection they're in? This is a bipartite match, and we want to estimate the average treatment effect on the treated.

This gives us ten matches, along with one extra for presence vs absence of any open business type.

For all elevent matches, we restrict our analysis to intersections with at least one business. This leaves us with $5,627$ of our $8,714$ intersections.

Average Opening Hours Differences in opening hours: how do opening hours affect crime? For each business type, we restrict our search to intersections with at least one business of that type that has opening hours.

For this match, liquor and lodging are not involved because the number of intersections with these types of businesses with recorded opening hours is too small.

This is a non-bipartite match: we use hours as a dosage, matching units that differ by at least five over the week, but by at most twenty. These are the minimum and maximum separation values, discussed in section 4.8.

Proportion Differences in proportion: for intersections with at least two businesses, does the proportion that are of each type matter?

This is a non-bipartite match: we use proportion as a dosage, matching units that differ by

[3] Total crime tends to be highly correlated with non-violent crime.

at least 0.05 over the week, but by at most 0.2; again these are the minimum and maximum separation values.

For this match, liquor is again not involved because the number of pairs formable with the above required difference in proportion is too small.

5.2 Results

We plot and discuss the results of the full matching procedure for each of our matches detailed in the previous section. In the following section, section 5.3, we will discuss secondary and sensitivity analyses, to investigate how reasonable our matching procedure results are.

Once we have a match, our outcomes of interest are the mean differences in crime, the standard error of that mean, and some function of how we chose the match in question.

The mean is easy to generate: it's just the average of the crime in the treated minus the crime in the control. In the non-bipartite matches, we scale the differences before taking means, as we will discuss below. Due to matching with replacement, we must use the method from section 4.5.4 to produce a valid standard error. This gives us t values, which give us p-values, that we Bonferroni adjust due to looking at multiple outcomes.

The mean effects given are for the entire period measured: for example, if the mean crime difference prior to data collection for a given business type was -7, this means that we estimate that business was associated with seven fewer crimes per intersection it exists at for the seven and a half years we measured. We only achieve causality if all assumptions are believed, including relevant time ordering[4].

In this section, the function of our match selection procedure of most interest is the number of pairs remaining after the procedure, compared to the total number possible. We will discuss many other aspects of the match selection in the next section.

[4]While in theory part of strong ignorability, we feel it's worth separating due to concerns of potential causal directions

5.2.1 Presence

For each business type, we pair intersections with that business type present to intersections with that business type not present[5]. Once we've run through our matching procedure, we produce a simple test of differences of crimes of each type between the treatment and control units, both for crime before collection of our business data, and crime after.

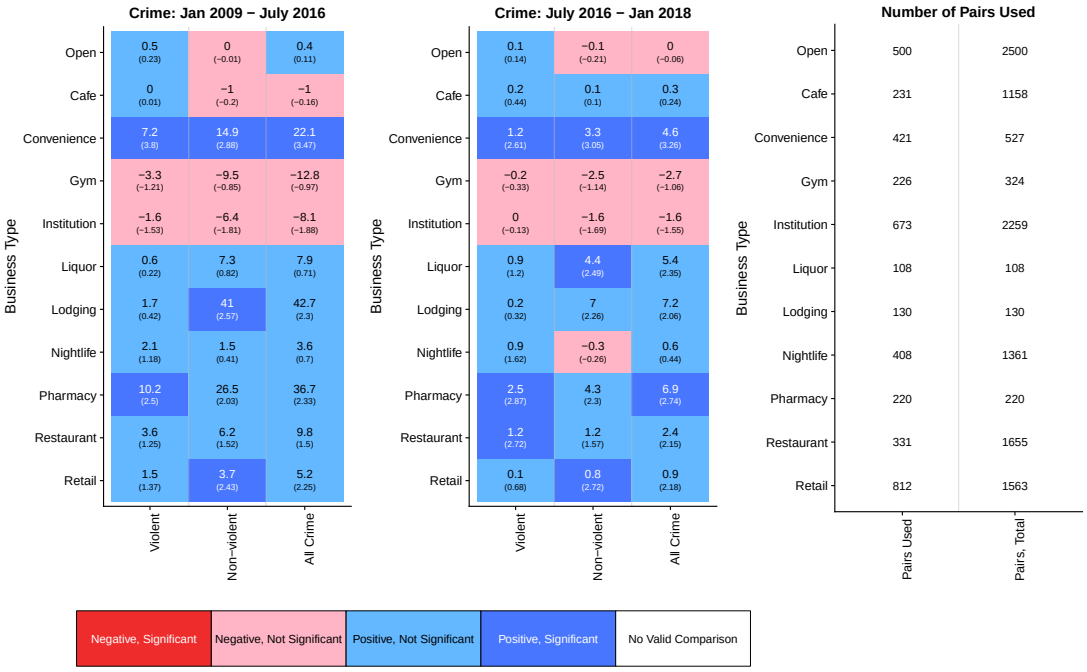


Figure 5.1: Matched pair mean differences: Presence of each business type vs Absence. We plot the mean crime differences, with the t values underneath. The boxes are coloured according to Bonferroni adjusted significance values. Here, positive means more crime is associated with presence of the business.

Figure 5.1 gives the results. We plot the results separately for crime before data collection and crime after, as discussed in section 2.3.2. The primary number in each box is the mean crime difference over the time period. Positive differences are blue: in these cases, crime is more associated with the treated intersection than the untreated; negative differences are pink or red. The significance of the relationship affects the colour, as detailed in the plot legend. The t-values of the test against zero are also given under each mean difference.

[5]But at least some business is present at the control intersection.

Finally, on the right, we plot the number of pairs used in each match, and the maximum number of pairs that could be used. For example, we used 812 pairs for the retail match. This means we used 812 unique intersections each with a retail business present, and matched them with up to 812 control units[6]. We formed matches using pair counts from 1,563 on downward, but all matches with more than 812 pairs were too predictable. For e.g. lodging, we could use all 130, because the best match formed was deemed unpredictable enough.

The majority of businesses are related with crime, i.e. the presence of the business type is associated with crime.

Institutions and gyms provides a consistent negative association. While not significant at the 0.05 level[7], the t values for institution were close to 2 for both violent and non-violent (and total crime), both before and after business data collection, with the exception of a small effect for violent crime after, although this is the smallest sample size comparison of the six.

Cafes and nightlife showed mostly null results, with a slight lean towards a positive association for nightlife.

Convenience, liquor, lodging, pharmacy, restaurants and retail showed versions of significant association with crime, both pre and post business data collection.

Note that the absolute effects are not huge in most cases, although this depends on perspective. Every convenience location is estimated to be associated with an increase of 7.2 violent crimes from 2009 till July 2016. That's about one a year at each of the 421 convenience locations. A similar effect size in terms of crimes per year is measured in the period after collection too.

Presence vs absence for open businesses showed mostly null results.

This is relatively consistent with the results from the core urban analysis, 3.3.2.

[6] As in, we allow matching with replacement, so we don't necessarily use 812 unique controls. The number of controls affects both the predictability of each match, and very directly the standard error.

[7] Bonferroni adjusted, like all comparisons here.

5.2.2 Average Opening Hours

For each business type, we pair intersections that both contain at least one business of that type with recorded opening hours, but such that one unit has between five and twenty more open hours per week.

These matches are of the "derivative" form referenced in section 4.8.2: after we form the match, the outcome is actually the difference in crime counts divided by the difference in opening hours. Thus if a pair of retail businesses are matched, one with 80 hours a week and the other with 65, we'll divide the difference in crime counts by 15. Thus, our outcome is expected crime increase per extra hour open.

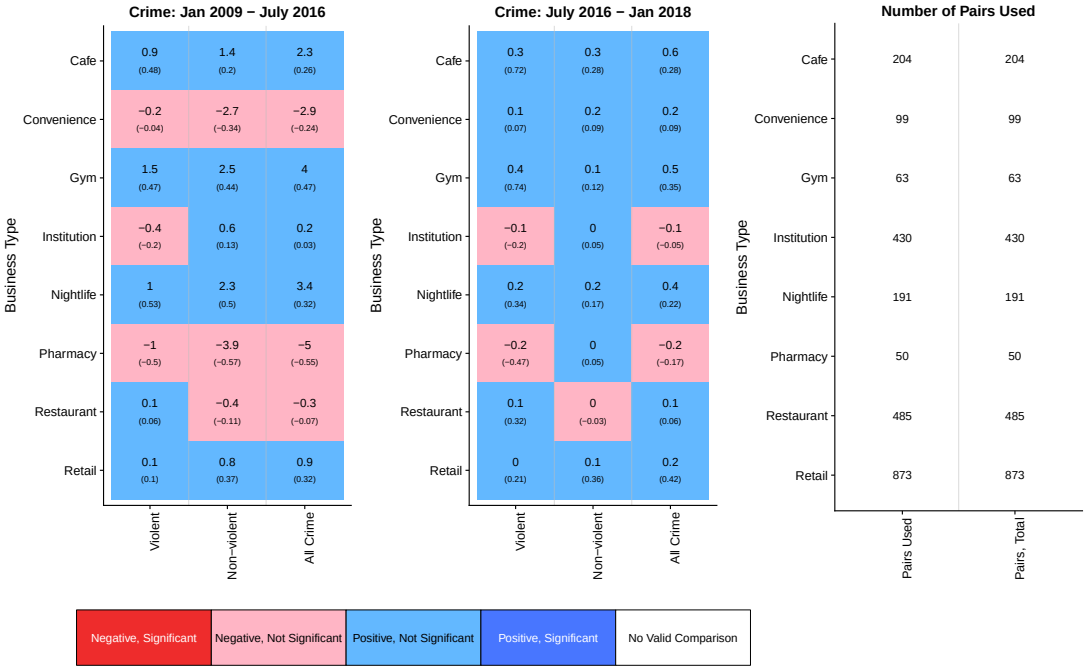


Figure 5.2: Matched pair scaled differences: Pairs of each business type, with differences in opening hours between of between five and twenty. We plot the mean scaled crime differences, with the t values underneath. The boxes are coloured according to Bonferroni adjusted significance values. Positive means more crime is associated with longer opening hours.

Figure 5.2 gives the results. No results at all were significant for differences in average opening hours. Some even had large sample sizes, including retail at 873. Pharmacies had

the largest absolute effects: every extra open hour is associated with five fewer crimes over the seven and a half years prior to business data collection, at each intersection with a pharmacy. Gyms had the strongest positive association between crime and opening hours.

We'll note that intersections with businesses with opening hours within a specified type look very similar to one another: every single business type found an appropriate match while using all treated intersections. That does not mean every possible match formed was deemed sufficient, only that the best match at $n = N_t$ was good enough.

These results are difficult to compare to the core results from section 3.3.1 and section 3.3.2, mainly due to the lack of significance in both.

5.2.3 Proportion

We restrict to intersections that both contain at least two businesses. For each business type, we record the proportion of business that are of that type. For example, if an intersection has eleven businesses and five are retail, the retail proportion will be $5/11$; if that same intersection has no cafes, the cafe proportion will be zero.

We then form pairs such that the difference in proportion is between 0.05 and 0.2; the treated unit is the unit with the larger proportion[8].

These matches are also of the "derivative" form, thus after we form the match, the outcome is actually the difference in crime counts divided by the proportion difference. Thus if a pair of intersections are matched, one with two cafes and seven total businesses and the other with one cafe and six total businesses, we'll divide the difference[9] in crime counts by 0.19. Thus, our outcome is expected crime increase per extra unit change (or 100%) of business of that type. This isn't truly meaningful, but we can divide by e.g. ten to get increase per ten percent etc, and the directions and significances don't change.

[8]This means for example that a zero proportion intersection for some type can only match to intersections with at least five businesses, as any proportion above 0.2 can't be matched to it. Also, intersections with one cafe and two total businesses (so one other) could match with intersections with one cafe and three total, but not with four total.

[9] The proportions are $2/6$ and $1/7$, so the difference is $2/6 - 1/7$.

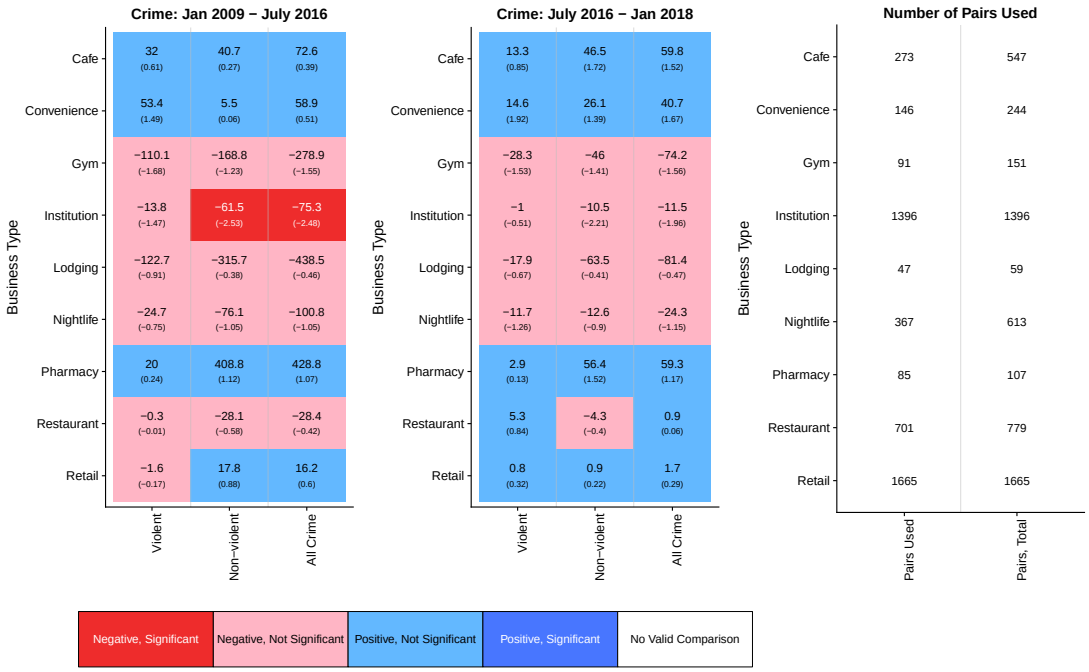


Figure 5.3: Matched pair scaled differences: Pairs of each business type, with differences in proportion between 0.05 and 0.2. We plot the mean scaled crime differences, with the t values underneath. The boxes are coloured according to Bonferroni adjusted significance values. Positive means more crime is associated with longer opening hours.

We'll note that these matches are correlated with presence vs absence. Because we match on number of businesses, these results will be somewhat zero sum.

Figure 5.3 gives the results. Institution is strongly and significantly associated with less nonviolent crime, and is associated with less violent crime. It is the only significant result for proportion.

Gym, lodging, and nightlife are consistently associated with less crime, although not significantly. Cafes, convenience locations and pharmacies are consistently associated with more crime, although not significantly.

Retail and restaurant have close to null results.

These results do not have a clear counterpart in the core analyses.

5.3 Validating Matches

The previous section does not tell us much about how our matches performed. Here we analyse specific matches, both to learn about the results on our data, and how our methods perform.

5.3.1 Analysing a selected Match in Detail

For a given match, that is, after we've gone through the entire process of testing all matches and number of possible matches and selected one match, we want to make sure it's a decent match.

We study many aspects of the given match:

1. We probably want to know what weight vector was selected: what weights for each vector ended up producing a highly unpredictable match?

2. As is always of interest, how well balanced, marginally, are the variables? This is similar but not identical to considerations in section 4.3.3: Mahalanobis imbalance, weighted or not, is all well and good, but it's useful to put things back on a more standard level for interpretation and robustness. We think about this in two ways. First, are the treated and control groups very different? This can be analysed with standardised differences: the mean difference in the two groups divided by the standard deviation of the variable[10]. If these are small, our match is good.

   Secondly, how well are the variables matched compared to random matching? That is, we form random pairs that obey the treatment requirement (and the min and max separation requirement in NBP) that formed the actual match, but with no consideration to the covariates. These random matches form a baseline to compare against. For bipartite matching, this is equivalent to asking how balanced are the

[10] Or rather, $\sqrt{2}$ times the standard deviation: $\mathrm{Var}(X-Y) = \mathrm{Var}(X) + \mathrm{Var}(Y) - 2\mathrm{Cov}(X,Y)$. When X, Y are independent with the same variance, we should get the standard deviation of the difference equal to $\sqrt{2}$ times the standard deviation of the variable.

variables compared to balance in the two groups before matching? This is not the same question as standardised differences.

3. In terms of Mahalanobis distance, i.e. attribute or covariate distance, how close are the pairs compared to a completely unrestricted match? That is, if we form an NBP match with Mahalanobis distance but don't enforce any treatment requirment, how much worse is our match? This tells us about how much of a restriction our treatment restriction really is.

   Similarly, how close are the pairs in physical distance, compared to the unrestricted match? Are they forced to be much further apart geographically than if no restriction has to be accounted for?

   Note that in both these cases, we don't use the treatment vector, but we do use the same set of pairs as a base pool. Thus for example comparing average hours, the NBP match can also only pull from all intersections with at least one business of the type in question with opening hours.

   In most cases, we allow the unrestricted comparison to throw away ten percent of the available match pool, for a more reasonable comparison.

4. The results: these are already given in the prior section, but we add a step: we also pull the fifty next best matches according to the permutation brier and Mahalanobis imbalance, and plot the outcomes from these.

5. Match details: Firstly, we include the same information as in the previous section: how many pairs were used, and how many pairs could have been used? Or of course the same questions another way - how many treated units were dropped until the match was deemed acceptable?

   What ratio was selected? This refers to the Mahalanobis ratio from section 4.5.3. What this tells us is how close to a Mahanalobis match we were, and how close to the propensity match. A value close to one means we selected a match close to a

Mahalanobis match; a value close to the cut-off ($R_{cut} = 2$ for us) implies the procedure tries to select matches close to the propensity match.

Finally, how did the match before in terms of Brier score? We record the selected Brier score, and its associated Brier p-value (two-sided). For extra clarity, we add a small plot of all permutation Brier scores with the selected score added, to see how it performed.

The first two items are analysed in one plot, as we'll see below. The x-axis contains the variables, grouped as described in 5.1.1, with the missing variables gathered at the end. The size of the points will correspond to the weight of the variable, and below the name of the category, we'll write the total category weight[11]. The y-axis will be the standardised differences. We never have to plot beyond 0.2 in either direction, as all our matches do a reasonable job of marginal balance. The points are finally coloured according to a comparison with a random match, or equivalently base differences in the case of bipartite matches.

We'll now look at three matches in detail.

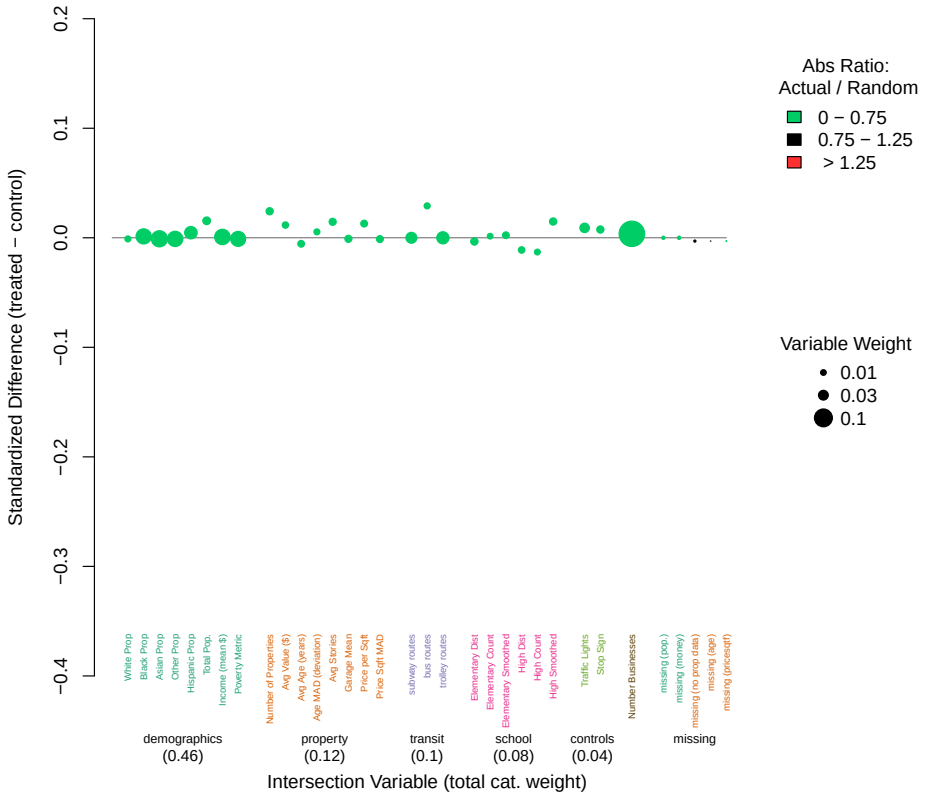
Restaurant: Presence vs Absence

Figures 5.4 and 5.5 plot the output described above in the case of presence vs absence of restaurant as the treatment.

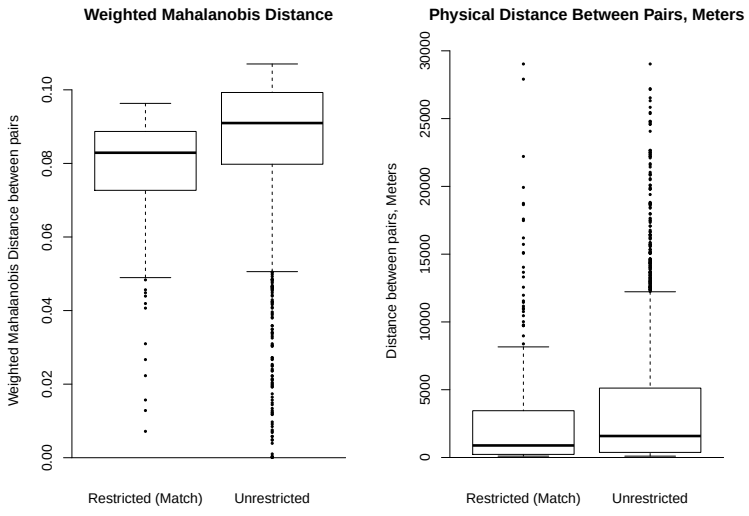
In the balance and weight vector plot, Number of Businesses has the largest individual weight, although less than half demographics as a category; a large weight was given to balancing demographic variables. The variables all have well balanced differences in the treatment and control groups, as all the standardised differences are very small. Further, except two of the missing variables, every variable is better balanced than in the random match comparison.

Below that, we do even better than the unrestricted match in attribute distance: this is

[11] Since the "Number of Businesses" forms a single category, its given weight is a variable weight; all the rest are sums of weights.

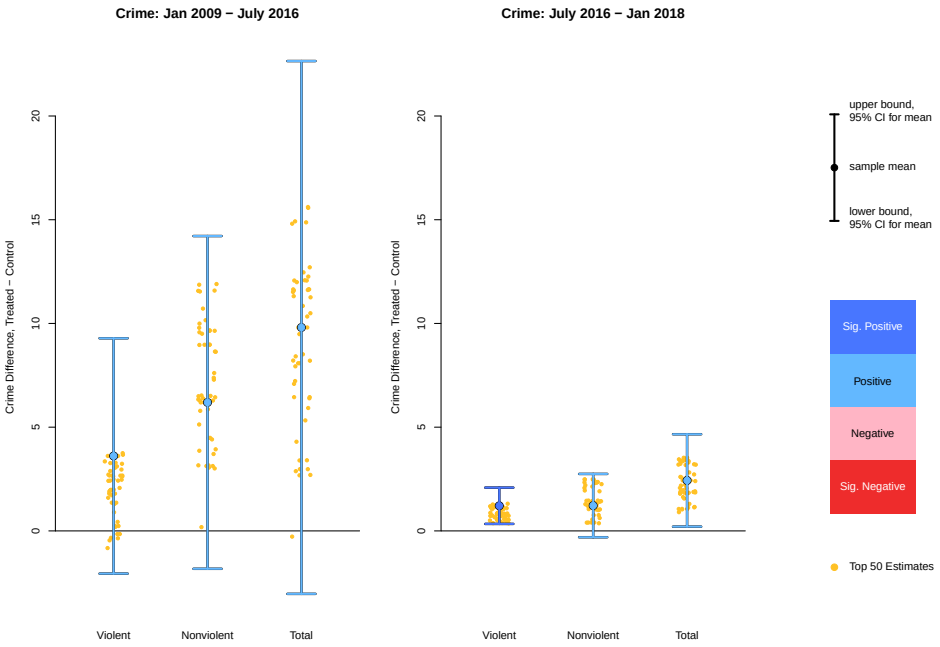


(a) Balance and Weight Vector

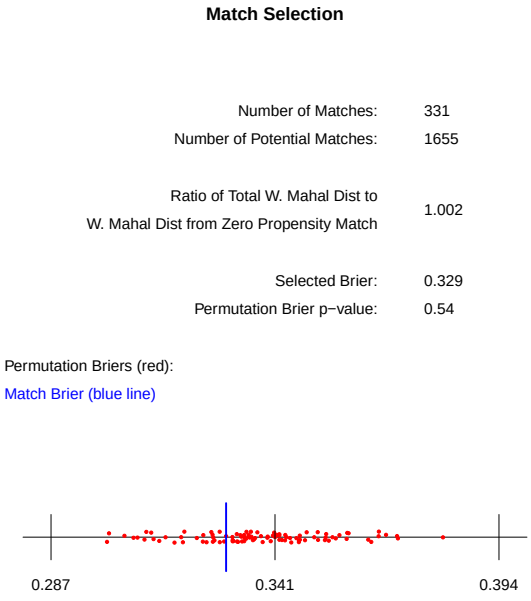


(b) Distance: Attribute (left) and Physical (right)

Figure 5.4: Restaurant match, Presence vs Absence. Match Validation part one.



(a) Crime Outcomes



(b) Selected Match Info

Figure 5.5: Restaurant match, Presence vs Absence. Match Validation part two.

because we only formed 331 pairs, while the unrestricted match had to form the full 1,655 pairs. But still, it shows that in this setup, our pairs really are now close together. They are also closer in physical distance than the unrestricted comparison pairs.

We've already seen the crime results. Generally, the top 50 comparison matches are not far off our best, and thus we have some confidence that our method does not have extreme variance. The fact that the standard error of our best match is on a similar scale to the variance seen from the fifty best match estimates gives an indication that those fifty best are not overly correlated with the best. As discussed in section 5.2.1, restaurants seem associated with crime.

The final box shows us again that we used 331 treated units of 1,655 possible. The ratio of 1.002 means this match was very close to a Mahalanobis match. The final part of this plot shows us how the selected Brier compares to the permutation distribution.

Convenience: Average Open Hours

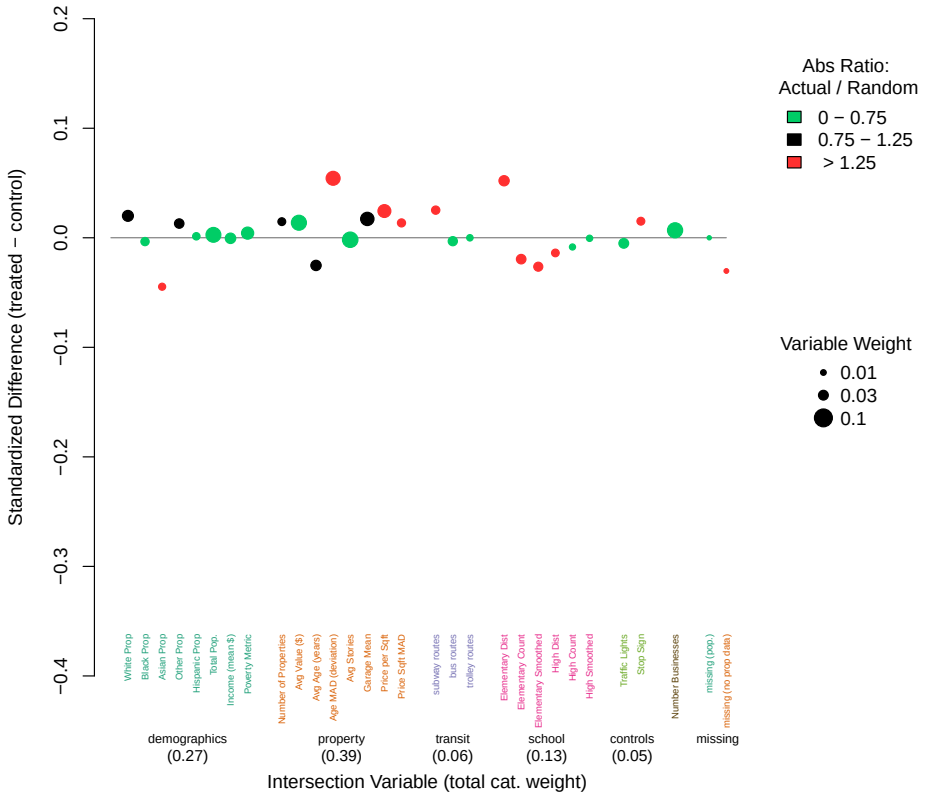
Figures 5.6 and 5.7 plot the output described previously in the case of differences in convenience stores opening hours as the treatment. Note that the set of possible units are the 198 intersections that contain a convenience location with opening hours.

We see some indicators that back up what we discussed in section 5.2.2. Firstly, we see that we don't do a whole lot better than random matches in balancing the covariates: this implies that the covariates are already reasonably balanced between the two groups. Considering we have 99 pairs in the match, the standardised differences are not too bad overall.

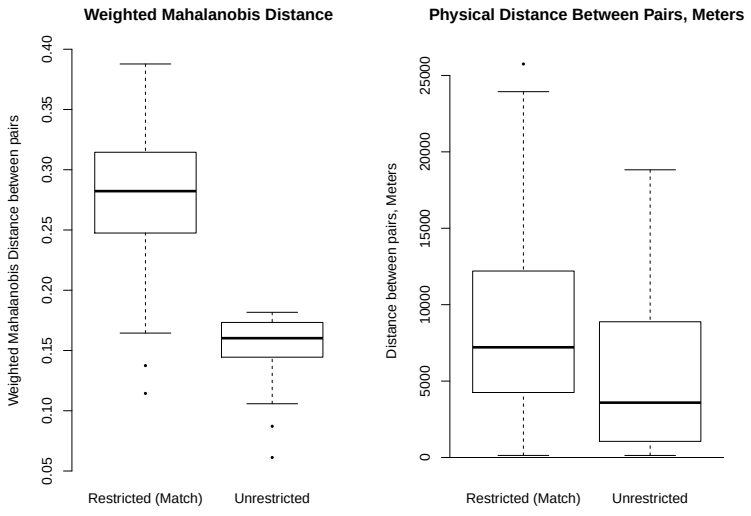
We do much worse in distance than the unrestricted pairings. One reason for this is that a restriction on 198 units is a tough restriction, relative to being able to form any pairs.

The noise in the outcome is larger than the variance in the fifty best matches. This indicates the fifty best are correlated with our best match.

The ratio of 1.384 shows that this match moved quite a bit towards the propensity match,

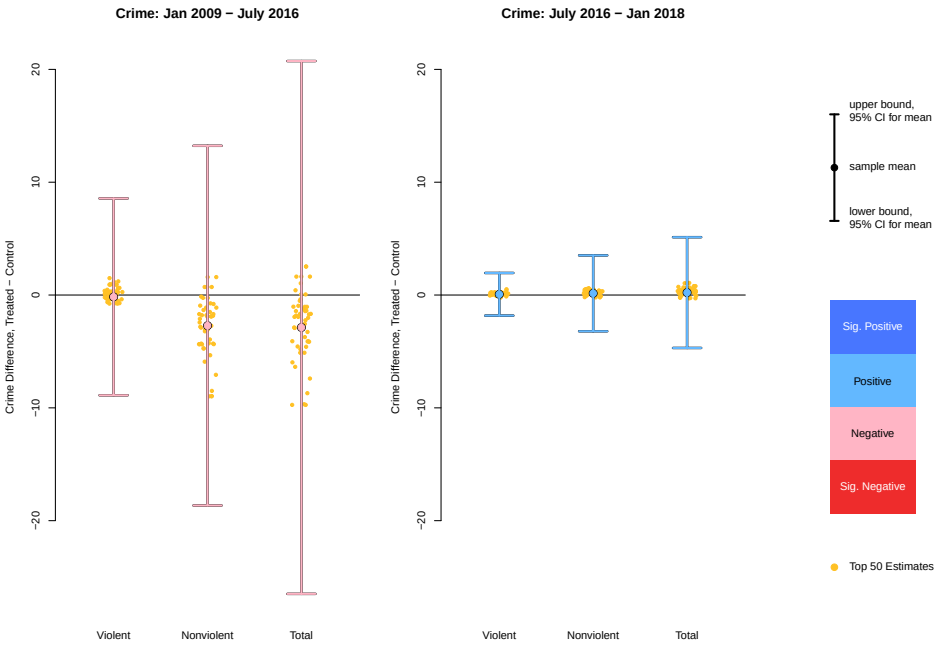


(a) Balance and Weight Vector



(b) Distance: Attribute (left) and Physical (right)

Figure 5.6: Convenience match, average opening hours. Match Validation part one.



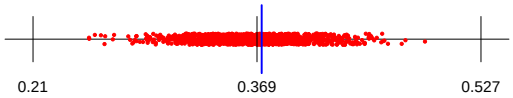
(a) Crime Outcomes

Match Selection

Number of Matches:	99
Number of Potential Matches:	99
Ratio of Total W. Mahal Dist to W. Mahal Dist from Zero Propensity Match	1.384
Selected Brier:	0.372
Permutation Brier p−value:	0.888

Permutation Briers (red):

Match Brier (blue line)



(b) Selected Match Info

Figure 5.7: Convenience match, average opening hours. Match Validation part two.

without being completely separated from the Mahalanobis match, in terms of calipers.

Institution: Proportion

Figures 5.8 and 5.9 plot the output described previously in the case of differences in proportion of institutions as the treatment. The set of possible units to use is large, as there are 1,396 intersections with at least two businesses with an institution.

A huge amount of weight is given to demographic variables, and very little to number of businesses. It might not surprise us to see much less weight on the number of businesses: we've already restricted to intersections with at least two, and we're operating on the proportion scale. The majority of variables are better balanced in the match than in the random matches.

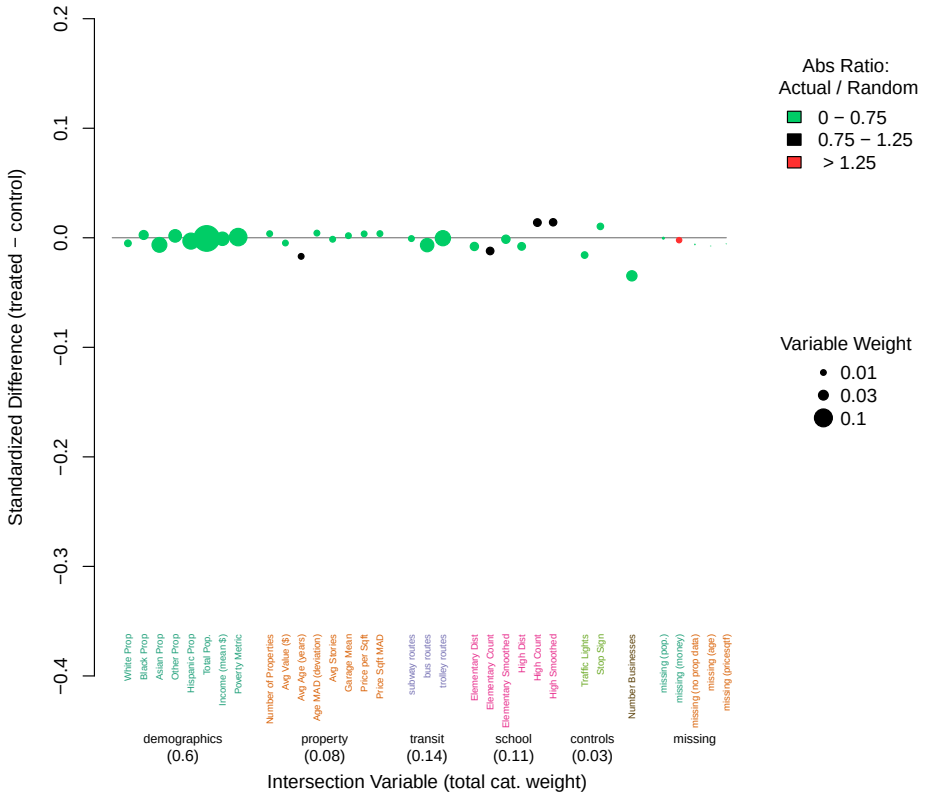
The only other thing worth noting is that we used all the pairs, but the balance and Brier scores were excellent, and the top fifty matches all produce about the same estimate as our best, with a similar spread. These are encouraging signs for a match.

5.3.2 Number of Matches Used

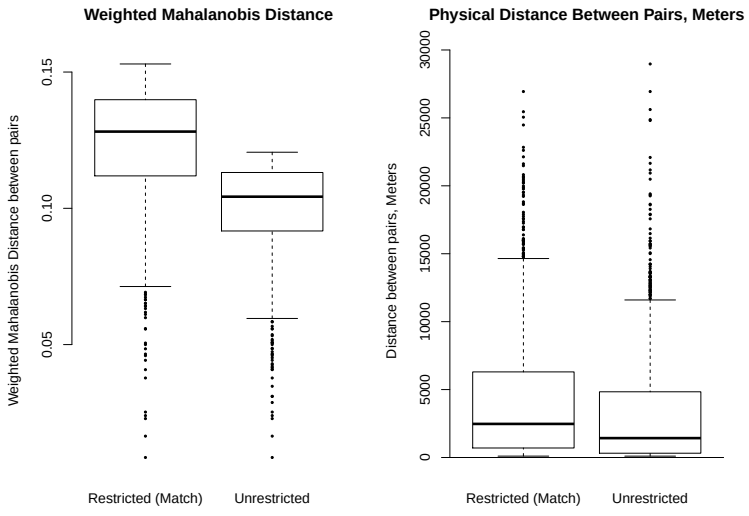
As detailed in our matching chapter, a large component of our matching method is that we keep reducing the number of matches used until our match is deemed sufficiently random.

We've just seen in the previous sections how we stopped immediately for the match on average hours and proportion: the best match selected at $n = N_t$ was already seemed good enough to use. However, for presence vs absence for restaurants, we had to go all way down from $N_t = 1,655$ to $n = 331$ before the match was unpredicable enough for our liking.

Looking at the best match for every n can be informative for our selection procedure. Firstly, is there a trend? As we reduce the number of matches and thus form closer matches, does the estimate trend in a particular direction? Does the variance shrink or grow? Of course as noted in section 4.7.3, these aren't all individually valid tests, and using such a plot to

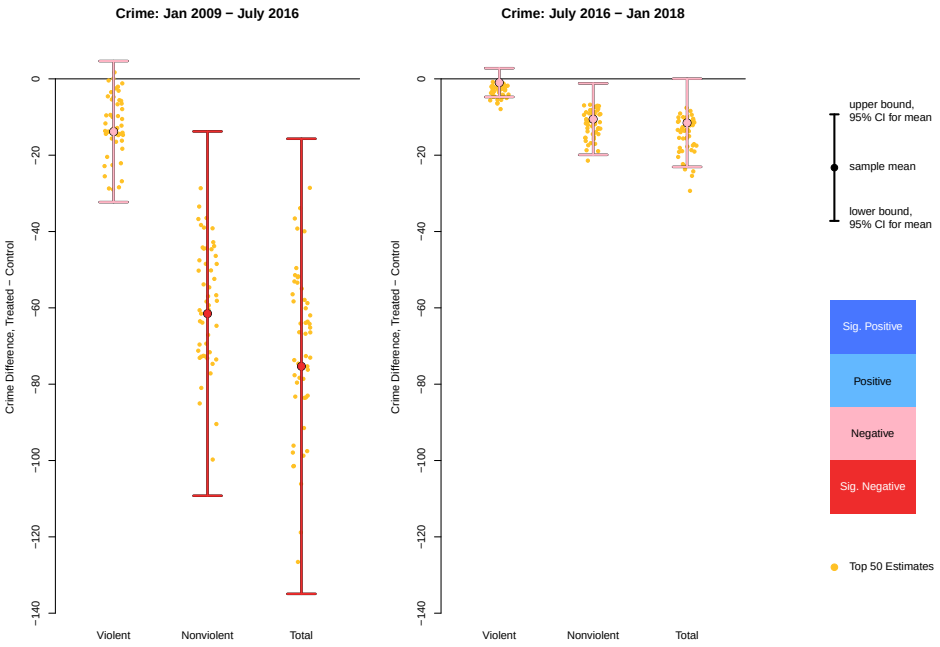


(a) Balance and Weight Vector



(b) Distance: Attribute (left) and Physical (right)

Figure 5.8: Institution match, proportion. Match Validation part one.



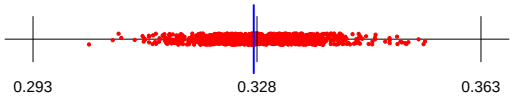
(a) Crime Outcomes

Match Selection

Number of Matches:	1396
Number of Potential Matches:	1396
Ratio of Total W. Mahal Dist to W. Mahal Dist from Zero Propensity Match	1.053
Selected Brier:	0.327
Permutation Brier p−value:	0.908

Permutation Briers (red):

Match Brier (blue line)



(b) Selected Match Info

Figure 5.9: Institution match, proportion. Match Validation part two.

choose a match is an egregious violation of statistical principles.

In the following plots, we will plot the mean and interval estimate of crime differences for every n we tested, i.e. the result of the best match at that number of pairs. One of these will be the interval we see in the detailed plots from the previous section, or similarly the overall plots from the results section. We colour each interval according to the Brier p-value: red if it's zero, thus every permutation Brier is larger; blue if it's sufficient, i.e. the Brier p-value is above the 0.05 cut-off; purple if it's in-between. Therefore the first blue interval we see is the one that we choose as our overall match, i.e. the match you see in the previous sections.

We do this for both violent and non-violent crime. We also add the regression line: it's the estimate of the treatment effect given all the data available to the match[12]. Further, we redo the regression at every n: whatever set of points the match uses, we run a regression on that set. This is matching as preprocessing.

We will plot four examples with discussion.

Number of Matches: Cafe, Presence

Figure 5.10 is the plot discussed above for matching on presence vs absence of cafes.

The most interesting component of this plot is the clear negative trend: as the number of pairs required decreases, the estimate tends towards zero. In fact, we stop at $n = 463$, the first time the match is sufficiently random. This is also the first time the difference is insignificant: for every "predictable" match, where the treatment and control groups are separable, we have a significant and positive effect. Variance isn't seemingly playing a huge role, so this is actually a trade-off of bias verses target: if the FATT isn't the same as the ATT, we are either converging towards the FATT, or we're decreasing bias due to poor matches - or of course potentially increasing bias due to an unobserved confounder.

The earlier poorer matches are above the overall regression line, and the later better matches

[12] So e.g. in the average hours comparison, we enter all the intersections with opening hours for a given business type to the regression.

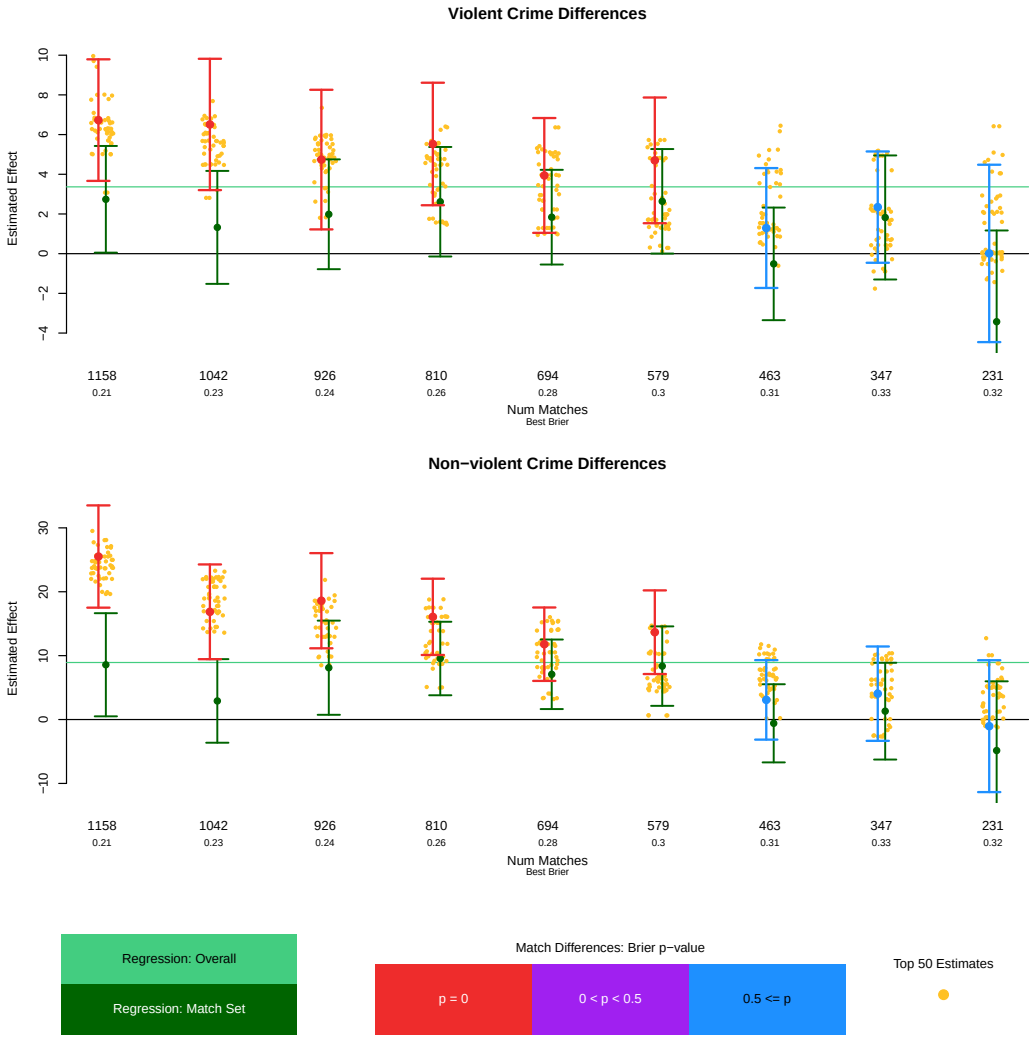


Figure 5.10: Crime Differences: Presence vs Absence of Cafe, by number of matches. The x-axis contains the number of matched pairs along with the Brier score for that match, i.e. the small number below the match count. The colours of the 95% estimates are: red for fully below the permutation brier distribution, purple for above the minimum, and blue for above the 0.5 cutoff. The light green line represents the overall regression estimate, and the dark green 95% intervals at each number represent regression estimates using matching as pre-processing. Violent, then Non-violent.

are below it. Further, we note that the variance of the matches does not seem to increase until we get to the smallest n used.

Both the top fifty estimates, and the regression as pre-processing estimates are consistent with the standard difference estimates.

This plot is for prior-business data crime: crime from 2009 till July 2016. The same pattern holds for post-business data crime.

Number of Matches: Restaurant, Presence

Figure 5.11 is the plot for matching on presence vs absense of restaurants.

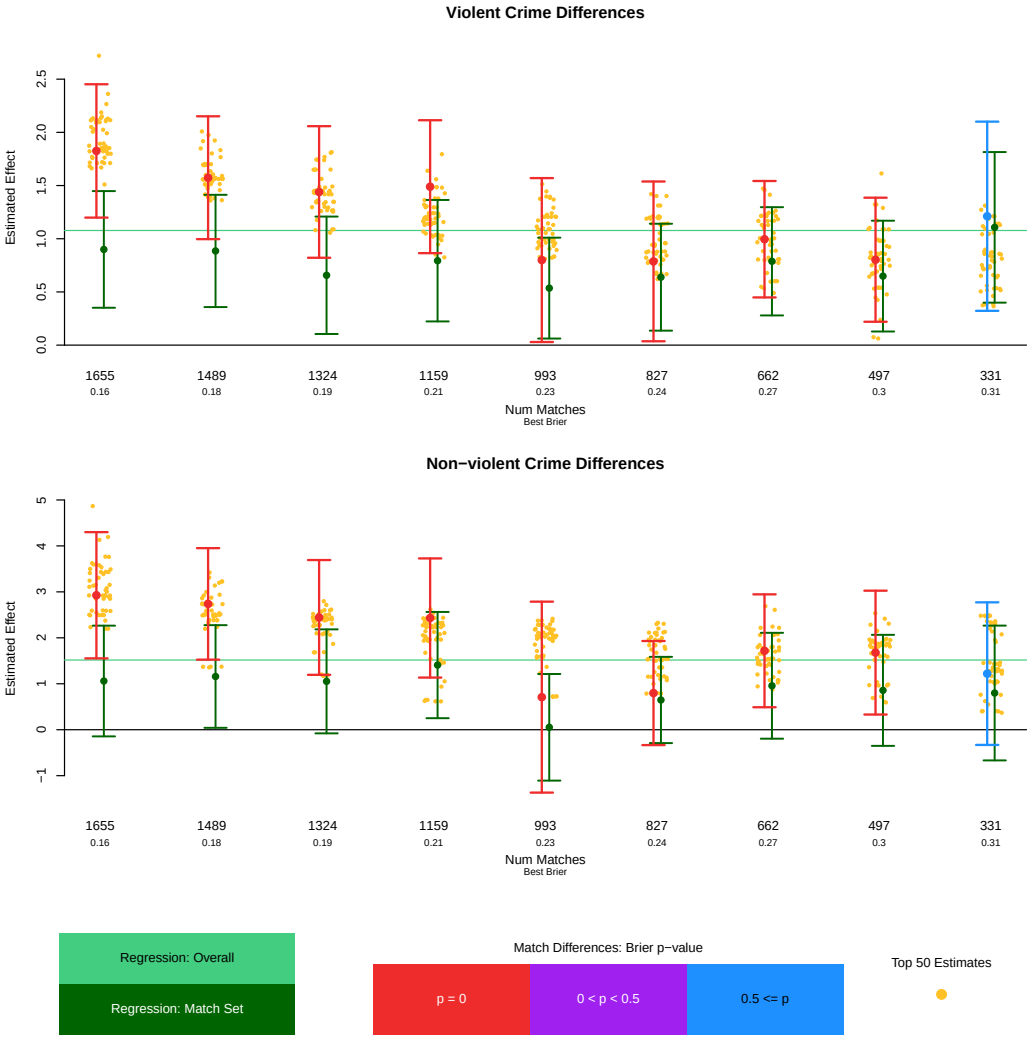


Figure 5.11: Crime Differences: Presence vs Absence of Restaurant, by number of matches.

This result is somewhat similar to the cafe result, except at least for violent crime, the result stays significantly positive, at every number of pairs down till we have an acceptable match.

This result is for post-business data crime, but much the same pattern holds for prior-business data crime.

Number of Matches: Retail, Average Hours

Figure 5.12 is the plot for matching on opening hours of retail businesses.

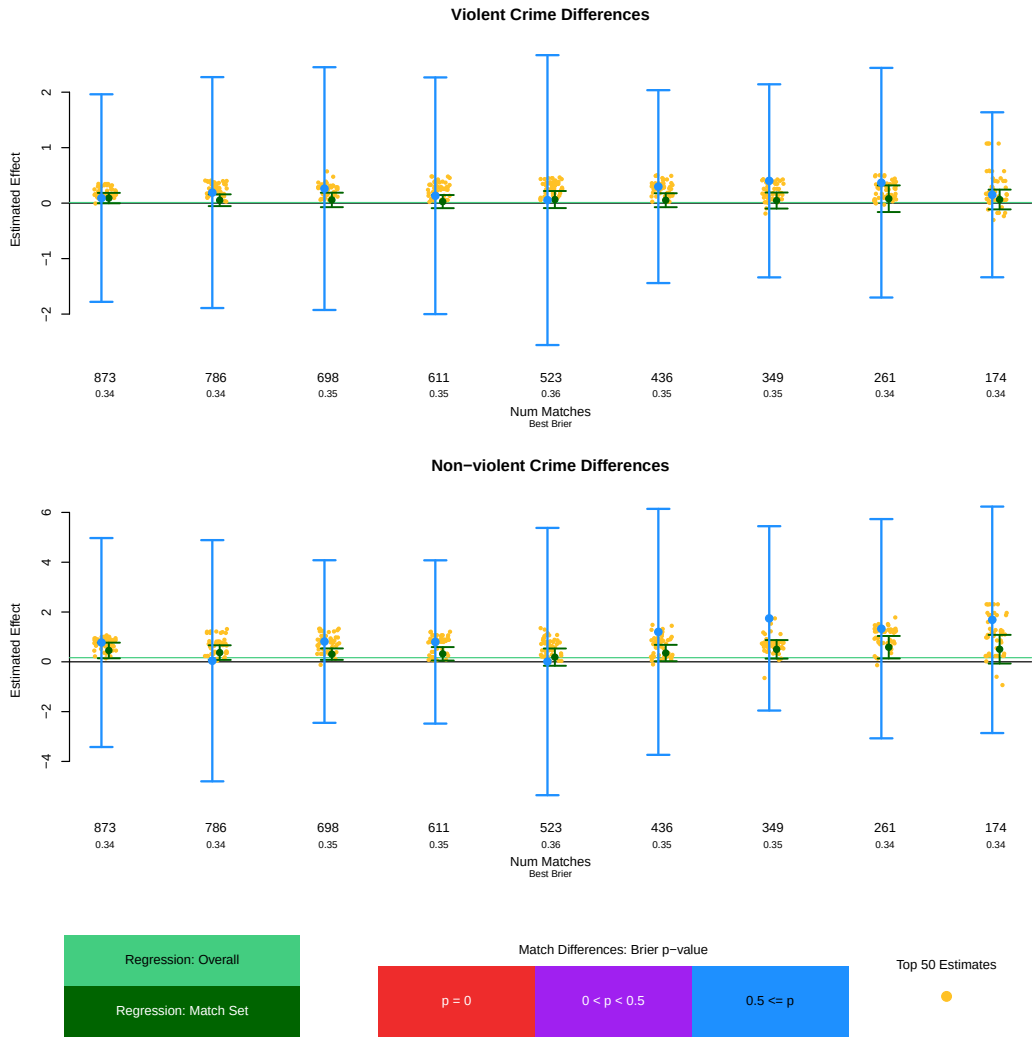


Figure 5.12: Crime Differences: Retail, opening hour differences of retail, by number of matches.

While this is a particularly extreme example, most of the average hour matches have very high variance. Part of that is a function of the minimum and maximum separation values

chosen, although it's worth noting that the regression estimates are not significant either.

All n values give good unpredictable matches, hence we stop right away.

This is yet another example of the fact that we often don't have a bias-variance trade-off: as we reduce the number of pairs, we often get smaller variance of the standard error. Here, the smallest number of pairs, $n = 174$, produces a smaller standard error than all other n values, for violent outcomes. This means for example the standard deviation of the largest set, $n = 873$, is more than $\frac{\sqrt{873}}{\sqrt{174}} = 2.24$ times the standard deviation of the smallest.

This plot is for prior crime, but again, we see much the same pattern in post crime.

Number of Matches: Gym, Proportion

Figure 5.13 is the plot for matching on proportion of gyms.

Finally we see some real variance growth, as n goes all the way down to thirty. In non-violent crime, we can see though that the variance grows for a while. Of course we stop at $n = 121$ in both cases, but it's worth noting that we suffer very small variance consequences even starting with small numbers of treated units, here 151.

While the match at $n = N_t = 151$ was too predictable, we see that very quickly the matches are hard to distinguish from the permutation distribution. This is the pattern noted in section 4.7.3: the procedure never spends much time with the Brier p-value in the interval $(0, 0.5)$: it's either below, or quickly jumps above as we reduce n. Only as we require extremely high p cut offs do we struggle.

This plot is for prior crime, but the pattern is very similar in post crime.

5.3.3 Crime Before and After: A Control

As mentioned in section 2.3.2, we split crime into two: crime data prior to business data collection, and crime data after business data collection.

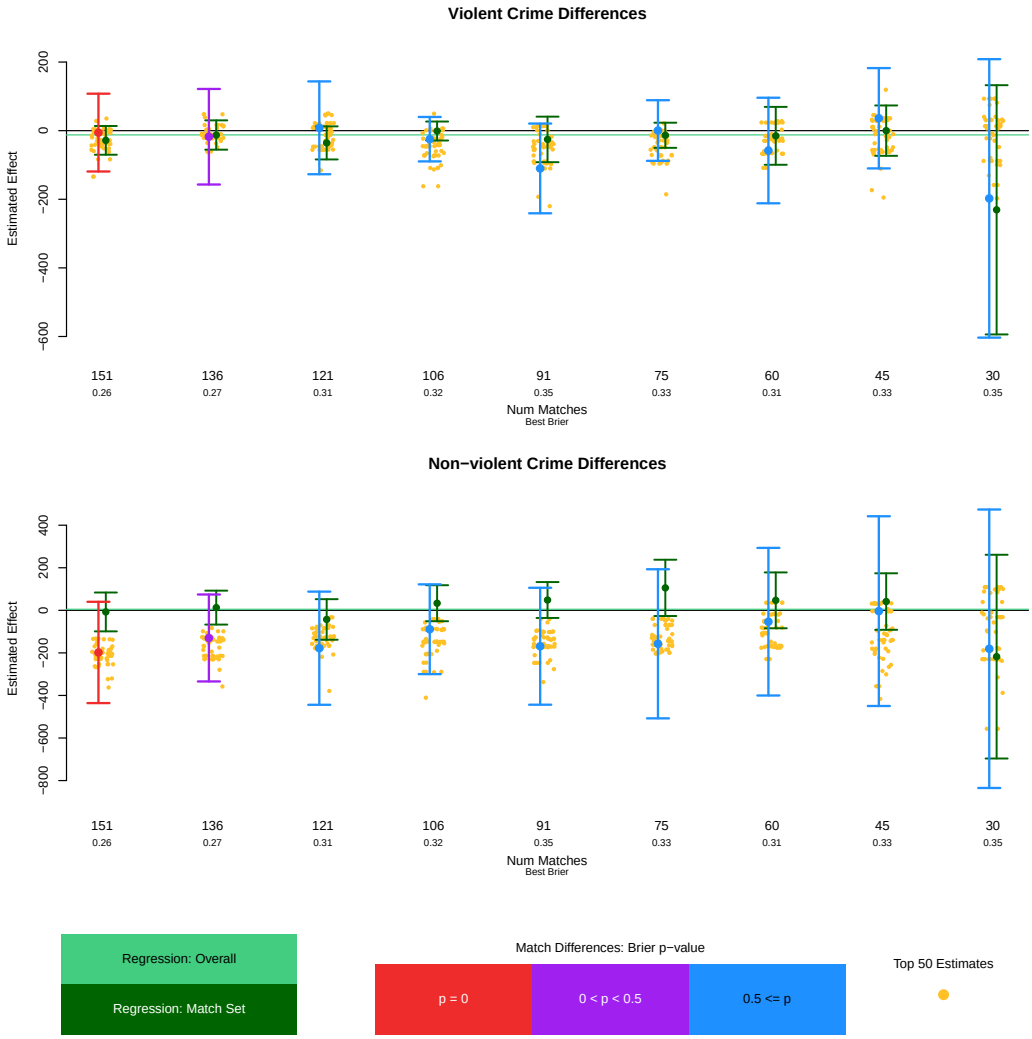


Figure 5.13: Crime Differences: proportion difference of gyms, by number of matches.

The idea is similar to that proposed by Pimentel, Small, and Rosenbaum (2016): building two control groups with different potential biases can rule out certain biases from fully explaining our results.

That is, if crime before data collection is a strong causal effect of business locations, then our work is backwards. Of course this is a possibility that we cannot rule out: crime could cause business, and business might not affect crime. The only issue is that businesses are generally positively associated with crime, thus this single causal pathway implies businesses build intentionally near high crime spots. Of course the analysis is more nuanced when it

comes to opening hours: it's reasonable that a business may adjust its hours downwards in response to rising crime, or increasing them in response to less crime. "Lucky" for us, our hours analysis was completely without significance.

Another criticism is that the businesses measured did not necessarily exist stretching back to 2009. This is a valid criticism, since we were not able to obtain accurate records of all 72,020 businesses. However, we imagine most were present for the 18 months after data collection.

There are endless possible ways for a prior analysis to be faulty while a post-analysis is not, or vice versa; there are probably even more ways for both to be invalid. Providing robustness in terms of consistent prior and post analysis at the least helps provide evidence against the first types of biases: biases that only affect one time period and not the other.

5.4 Discussion

We are left in a similar position as we were in at the end of chapter 3: crime happens near business, and the relationship with opening hours is complicated. The proportion analysis tells us something more about comparing highly built up areas. All told, this might be the most informative of all, but it's hard to shake the feeling that we are still just not capturing everything there is to know about an area.

Further, it's very hard to look at these results and arrive at anything resembling a policy conclusion. Matching is not at fault: perhaps the opposite; these well-matched sets avoid fooling us into generating bias-based policy. On the other hand, this does suggest that any rules of thumb about certain businesses being obviously good in terms of neighbourhood safety need further justification; the onus is even greater for claims about opening hours.

Our hope is that our methods and data can provide a pipeline for researchers with potentially better data and ideas with a solid base to work from, to push urban analysis even further.

Chapter 6

Modelling Lottery Incentives on Daily Adherence

6.1 Modelling Lottery Incentives: an Introduction

6.1.1 Adherence to Daily Activities

Many chronic health issues require daily (or multiple times daily) adherence to medication for optimal management; examples include diabetes, hypertension, and hypercholesterolemia. Some conditions, such as obesity, can be improved with daily physical activity. Other issues, such as addiction or substance abuse, are treated with abstinence programs in which the daily "activity" is recorded non-use of the substance.

Many of these health-promoting activities can be defined as adherence to a daily task. If medications are to be taken once or more per day, adherence constitutes taking all of the required pills for the day; failure to take any portion would constitute non-adherence. Adherence to daily activity can be defined as reaching a specified target, such as 7,000 steps walked per day. In abstinence studies, adherence can be defined as achievement of an abstinent day.

The purpose of defining health behaviors as adherence to a daily task is twofold: it permits both monitoring of activity and incentivizing such activity. While adherence can be defined for any period, longer periods make it difficult to decipher short-term behavior effects, and make individual daily behaviors less salient. On the other hand, too-frequent monitoring (e.g., twice-daily pill taking) may require burdensome evaluation and feedback, causing a disassociation between adherence and intervention.

6.1.2 Incentive Mechanisms

There are a variety of ways to structure financial incentives, including as fixed payments, daily lotteries, pre-commitment devices such as deposit contracts, or with non-monetary prizes. We focus here on daily lotteries in which "winning" is conditional upon fulfillment of the targeted daily activity. These lotteries incorporate several powerful concepts derived from behavioral economics, a field that incorporates both economic principles and insights from psychology to effect good decision-making and positive behavior change. For example, we tend to misinterpret small probabilities, a phenomenon that may explain the popularity of state lotteries with very low expected values. In addition, we experience "loss aversion," in which the loss of a certain size is more distressing than a gain of equivalent size is reinforcing (Tversky and Kahneman 1991). We also experience "regret aversion," in which the emotional cost of regret (e.g., having missed the chance at a reward) is significant (Loomes and Sugden 1982). Using these concepts, we have designed "regret lotteries" that take advantage of many of these concepts to encourage desired behavior. These are described in more detail below.

In this paper, we aim to model the lottery program's effect on daily adherence, in the context of adherence to a daily medication regime. We model daily adherence, as a function of the daily lottery outcomes. Our main goal is to understand the mechanism of the lottery, and how it affects both short- and longer-term adherence. We wish to form hypotheses about future lottery incentive structures, including how to best allocate a fixed amount of money[1].

6.1.3 Binary Time Series

The analysis of autocorrelated time series with binary outcomes is less straightforward than analysis for the continuous equivalent, as we cannot apply well-developed Gaussian methods. In place of autoregressive integrated moving average models, binary models can use generalized linear autoregressive integrated moving average models; these models are

[1] More specifically, a fixed expected amount per adherent day: the lottery naturally adds a random element to payouts, and in all payment structures, payouts grow linearly with adherence, as is desired.

referred to as observation-driven, because the distribution of the outcome at a given time t depends explicitly on prior observations, and not on a hidden process.

In contrast to observation-driven methods, "parameter-driven" methods incorporate a latent process to account for dependence. Kalman filtering is an example in continuous settings, and more generally we have hidden Markov models and dynamic Bayesian networks. It is common to assume a discrete hidden structure to underlie a discrete time series, but for many applications, including this paper's application, discrete hidden states do not offer much inferential benefit over observation driven methods. Similar to Wu and Cui (2014), we will assume a continuous underlying process.

The resulting model has useful inferential properties in its own right: we get both a sense of the underlying autoregressive structure and the directionality and significance of our covariates. Following Campbell and Stanley (1963), we then analyze our multiple time series as comparative interrupted time series, using the output of our regression models as the control mechanism for comparative time series.

6.2 Lottery Structure

In our trials, lottery incentive group members were first asked to choose a personal two-digit number between 00 and 99. Every day, a random number was selected as the winning lottery number. If a participant's number matched the lottery on one digit (18% chance), s/he was eligible to win a small amount; if the participant's number matched both digits (1% chance) s/he was eligible to win a large amount. The "win" amounts varied slightly by trial; generally the small prize was \$5 or \$10, and the large prize was \$50 or \$100, resulting in expected values of approximately \$1.50 or \$3.00, respectively. An important feature of the lottery is that if the participant received any winnings *only if s/he had been adherent the previous day.* This "regret" feature, along with the variable reinforcement produced by randomness in the frequency of winning as well as the magnitude of the prize, enhances the motivational strength of the lottery.

6.2.1 Medication Adherence and Hyperlipidemia

In the Shared Incentives study (Asch et al. 2015), to incentivize adherence to cholesterol-lowering medication (i.e., statins), there were four treatment arms: a control group, a physician incentive group (physicians received direct payments), a participant incentive group (participants entered in a daily lottery like that described above), and a shared physician and participant incentive group. To demonstrate our approach, we focus on the participant incentive groups here.

Both groups receiving participant incentives participated in a lottery as detailed in the previous section, with the participant incentive group receiving \$100 and \$10 for large and small wins, respectively, and the shared incentive group receiving \$50 and \$5. We include the shared incentive group for analysis as part of the "lottery group" and we ignore any effect of the physician incentives.

6.2.2 HeartStrong

The Heartstrong study (Troxel et al. 2016), designed to incentivize adherence to beneficial medications following a heart attack (i.e., statins, aspirin, beta-blockers, and anti-platelet medications) included a control and an incentive arm. The incentive arm received the same lottery as detailed above, with large win amounts of \$50 and small win amounts of \$5, along with support from a personal supporter and study-supported social worker.

6.3 Data Description

Our general problem consists of participants $i = 1, \ldots, N$, each with a time series $\{Y_t^i\}$ over a set study period $t = 1, \ldots T$ (due to start-up issues with study devices, T is often participant-dependent), with $Y_t^i = 1$ if the goal is completed, and 0 otherwise. Most of our

studies are of the form:

$$Y_t^i = \begin{cases} 1 & \text{pill taken on day } t \text{ by participant } i \\ 0 & \text{otherwise} \end{cases}$$

Or:

$$Y_t^i = \begin{cases} 1 & \text{participant } i \text{ walks} \geq 7000 \text{ steps on day } t \\ 0 & \text{otherwise} \end{cases}$$

For the case of medication adherence, a binary time series is the natural choice.

If the goal was completed on day t, then the participant is awarded the lottery winnings, and is informed. If the goal was not completed, the participant receives a "regret" message telling her/him that s/he *would have* won, if s/he had only completed the goal. We represent these outcomes with four indicators, with l and L referring to the small and large lotteries respectively, and w and r referring to wins and regrets:

$$\begin{aligned} l_w &= \begin{cases} 1 & \text{small win, Y = 1} \\ 0 & \text{otherwise} \end{cases} \qquad l_r = \begin{cases} 1 & \text{small win, Y = 0} \\ 0 & \text{otherwise} \end{cases} \\ L_w &= \begin{cases} 1 & \text{large win, Y = 1} \\ 0 & \text{otherwise} \end{cases} \qquad L_r = \begin{cases} 1 & \text{large win, Y = 0} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{6.1}$$

These are participant- and day-dependent, so we collect them to give $\mathcal{L}_t^i = (l_w, l_r, L_w, L_r)_t^i$. Note that most of the time (approximately 81% in the studies described here), $\mathcal{L}_t^i = (0, 0, 0, 0)$, i.e., no lottery winnings occurred.

The lottery results for participant i on day t are given after Y_t^i is recorded, thus our covariates for day t can only contain a function of $\{\mathcal{L}_1^i, \ldots, \mathcal{L}_{t-1}^i\}$, lottery results from days prior to t.

Assessing the total value of the lottery, defined in equations 6.2 (ATE during the study period) and 6.3 (ATE during the follow-up period) below is not our main objective. These are more properly compared with aggregation methods used in, e.g., Patel et al. (2016) and Troxel et al. (2016). Our goal is instead to analyze short-term response to the lottery, in order to understand the mechanism of the lottery, and the optimal design for future lotteries.

$$\text{ATE}_{\text{in-study}} = \mathbf{E}\left[\sum_{t=1}^{T_i} Y_t^i \mid \text{participant } i \text{ in lotto arm}\right] - \mathbf{E}\left[\sum_{t=1}^{T_i} Y_t^i \mid \text{participant } i \text{ in control arm}\right] \tag{6.2}$$

$$\text{ATE}_{\text{post-study}} = \mathbf{E}\left[\sum_{t>T_i} Y_t^i \mid \text{participant } i \text{ in lotto arm}\right] - \mathbf{E}\left[\sum_{t>T_i} Y_t^i \mid \text{participant } i \text{ in control arm}\right] \tag{6.3}$$

It is possible the entire effect of the lottery mechanism is non-responsive to messaging: that is, participants adjust their baseline goal-completion rate due to the knowledge of being in the lottery arm, and conditional on this have no response to daily messaging and payouts. It is also possible that the lottery could be beneficial in terms of the average treatment effect but produces a locally negative effect. For example, winning the lottery could make participants decide that they have earned a day off, leading them to be non-adherent the following day. If this is the case, then prior psychological knowledge, or running many lottery experiments of different types, would best inform lottery design. We will not be totally left in the dark in such a scenario, as we can compare local, or short-term, effects of the lottery with the ATE from aggregated models. Patterns found under such a comparison can still inform lottery design.

6.4 Matching and Modelling Daily Adherence

In this section, we outline our two-pronged approach for our matching analysis. Our main interest is to compare the difference in adherence between lottery winners and non-winners, and similarly, between those who received a regret message and those who did not. Once the

controls, i.e. the appropriate comparison group, are chosen, this method is straight-forward.

Choosing suitable controls is the most difficult aspect of most matching analyses, and the same is true here. We want to match people with an approximately similar base rate of adherence at the time of the comparison, else our differences in adherence rates won't be correctly estimating the effect of the lotteries. To this end, we cannot match people based on overall adherence: this is partly due to post-treatment matching bias (see e.g. Rosenbaum 1984), but also ignores the variance in adherence probability for each participant over the course of a study. We could also match on recent adherences; we compare this method to ours in section 6.7.4.

Instead we model the daily adherence probability for each participant. We use the output of this modelling procedure to form our controls for our matching procedure.

6.4.1 Data Structure and Latent Processes

We have N time series, $\{Y_t\}^i$, $i \in \{1, \ldots, N\}$. Each time series is a binary sequence, $Y_t^i \in \{0, 1\}$, $t = 1, \ldots, T_i$, corresponding to daily adherence. Each sequence has associated covariates, $\{\boldsymbol{U}_t^i\}$, and a parameter vector $\boldsymbol{\beta}^i$. The covariates vector includes a function of $\{\mathcal{L}_1^i, \ldots, \mathcal{L}_{t-1}^i\}$, i.e. all lottery results up to day $t-1$[2].

A standard generalized linear model assumes the mean of Y is a function $G(.)$ of $(\boldsymbol{\beta}^i)'\boldsymbol{U}_t^i$, where $G(.)$ is a function from $\mathbb{R} \to (0, 1)$, typically a CDF such as the logistic function, or the Gaussian CDF. The issue is the lack of independence: the unconditional mean $\mathbf{E}[Y_t^i \mid \boldsymbol{U}_t^i]$ is unlikely to be the same as the conditional mean $\mathbf{E}[Y_t^i \mid Y_{t-1}^i, \boldsymbol{U}_t^i]$. Note that for binary data, $\mathbf{E}(Y) = \mathbf{P}(Y = 1)$.

We can solve the correlation issue in multiple ways. From Cox et al. (1981), the two most general descriptions are observation-driven models and parameter-driven models. In observation-driven models, Y_t^i depends explictly on prior values Y_τ^i for $\tau < t$. See the GLARMA package (Dunsmuir, Scott, et al. 2015) for R. In parameter-driven models, we

[2]The lottery on day t is a function of Y_t, thus $\mathcal{L}_t$ cannot be a predictor for Y_t

assume a hidden state on which Y depends. In our opinion, parameter-driven models offer a more natural interpretation of the process of the time series. Under simulation, they project less bias onto future predictions. The disadvantage is in fitting these models.

We assume $G = \Phi$, the normal CDF, an underlying process X_t^i, and an autocorrelation parameter $\varphi^i \in (-1, 1)$ such that:

$$\Pr(Y_t^i = 1 \mid X_t) = \Phi(X_t^i)$$

With X incorporating the covariates and the autocorrelation:

$$\begin{cases} X_{t+1}^i = (\boldsymbol{\beta^i})' \boldsymbol{U}_{t+1}^i + \eta_{t+1}^i \\ \eta_{t+1}^i = \varphi^i \eta_t^i + \varepsilon_{t+1}^i \end{cases} \tag{6.4}$$

Or in one step:

$$X_{t+1}^i - (\boldsymbol{\beta^i})' \boldsymbol{U}_{t+1}^i = \varphi^i \left(X_t^i - (\boldsymbol{\beta^i})' \boldsymbol{U}_t^i \right) + \varepsilon_{t+1}^i \tag{6.5}$$

with ε_{t+1}^i being a zero mean, IID variable such that $\mathrm{Var}(\varepsilon_{t+1}^i) = \sigma_i^2$. Unconditionally, equations 6.4 give $\mathbf{E}X_t^i = (\boldsymbol{\beta^i})' \boldsymbol{U}_t^i$. It generally won't be necessary to assume that ε is normally distributed.

This model is equivalent to having an underlying process α_t, with $\mathbf{P}(Y_t \mid \boldsymbol{U}_t) = \Phi(\beta' \boldsymbol{U}_t + \alpha_t)$, with α_t being an autoregressive mean zero process with no covariates.

6.4.2 Structure of U and Decaying Lottery Effect

Our predictors, the $\boldsymbol{U}_t^i$ vectors, contain an intercept, a time variable, and prior lottery results. The time variable is to account for potentially changing base goal completion rates over the course of the study.

As detailed in our data description and equations 6.1, we allow the lottery to affect X in

four ways: when you win the lottery, and when you would have won if you had been eligible, i.e. when you receive a regret message; and for each, when the amount is large, and when the amount is small. We called this collection of mutually exclusive indicators $\mathcal{L}_t^i$, with $\mathcal{L}_t^i = (l_w, l_r, L_w, L_r)_t^i$.

Further, we allow these lottery effects to propagate beyond the next day, into the future. If the lottery only affected the next day, we'd have:

$$\boldsymbol{U}_t^i = [1,\ t,\ (l_w, l_r, L_w, L_r)_{t-1}^i] \tag{6.6}$$

Instead, we allow $\boldsymbol{U}$ to contain a function of prior lottery results. The method is detailed in the following section.Essentially $\boldsymbol{U}$ contains a power decayed function of the most recent lottery, with the rates of decay also parameters in our model. Each lottery effect is assumed strongest initially, but may continue to have some effect for future days. Our models allow both the shape of the decay and the length of the decay to vary. We fit the decay parameters separately for the large and small lotteries.

6.4.3 Decaying Lottery Effect

In allowing the lottery to affect future days, we setup a decaying structure on all four lottery effects. For the sake of identifiability, it's hard to justify separate decay parameters for winning and regret, so we fit the same parameters for the two large effects, large wins and large regrets, and a separate set of parameters for the small effects, small wins and small regrets.

We assume decay is parametrized by (γ, λ), i.e. we have some function $G(x, \gamma, \lambda)$ that gives the weight of values from x days ago. We assume that $G(0, \gamma, \lambda) = 1$ for all γ and λ, so that all decay is relative to day one.

γ can be thought of as the shape parameter, and λ the length parameter. λ is how many days the lottery lasts for, so that $G(x, \gamma, \lambda) = 0$ for any $x > \lambda$. γ controls how the effect

scales down to zero at $x = \lambda$: when $\gamma = 1$, the decay is linear; for $\gamma < 1$, the effect decays faster than linear, and for $\gamma > 1$, the values decays slower than linear, i.e. the effect of the lottery is stronger for longer.

For our specific purposes, we use the following functional form:

For our data, we assume:

$$G(x, \gamma, \lambda) = \begin{cases} \sqrt[\gamma]{1 - (x/\lambda)^\gamma} & x \leq \lambda \\ 0 & x > \lambda \end{cases} \tag{6.7}$$

While x will only ever be an integer for our purposes, this function is defined for all $x \geq 0$. λ can also be continuous.

We can then use the propagated lottery effects in $\boldsymbol{U}$. Note that we only propagate the most recent effect. Recall our vector of lottery effects, $\mathcal{L}_t^i = (l_w, l_r, L_w, L_r)_t^i$. Without loss of generality, let's focus on just one effect, say the small wins, l_w, on day t for person i.

If one of the other three results happened more recently, that is, if we won big, had a large regret, or a small regret more recently than a small win, we set $(l_w)_t^i = 0$. If small win was our most recent result, assume we won d days ago, with $d \geq 0$. We set:

$$(l_w)_t^i = G(d, \gamma_{\text{small}}, l_{\text{small}}) \times (l_w)_{t-d}^i$$

Of course, if $d = 0$, i.e. we won on day t, we set $(l_w)_t^i = 1$. Thus we can just eliminate the indicator from the above definition, to get:

$$(l_w)_t^i = \begin{cases} G(d, \gamma_{\text{small}}, l_{\text{small}}) & \text{won } d \text{ days ago} \\ 0 & \text{other lotto result since} \end{cases}$$

And indeed, if no lottery result has happened at all yet, we set $\mathcal{L}_t^i = (0, 0, 0, 0)$.

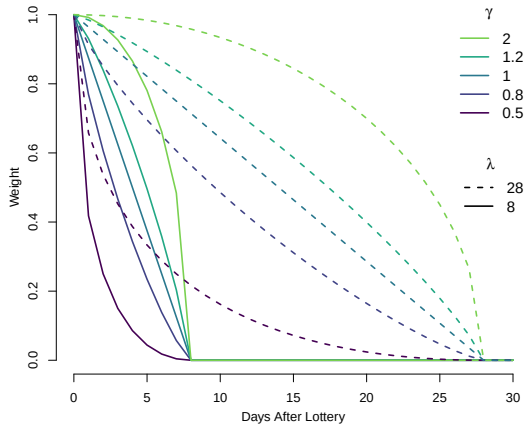


Figure 6.1: Decay curves for different γ and λ values

For the small lotteries (win and regret), recall we have $p_{\text{small lotto}} = 0.18$, and for the large, we have $p_{\text{large lotto}} = 0.01$. We're assuming a finite effect, with no effect after at most twelve days for the small lotteries, one hundred for the large; i.e. we limit λ_{small} to be less than 12, and λ_{large} to be less than 100. This allows a flexible decay pattern, seen in figure 6.1.

While we can have individual small lottery coefficients for participants, due to the large amount of noise in the data we assume a shared value of γ_{small} and λ_{small} over all participants, and similarly a shared γ_{large} and λ_{large}.

6.4.4 Modelling Methods for Regression

If primary interest is inference for the $\boldsymbol{\beta}^i$ vectors, we can follow Wu and Cui (2014) or Dunsmuir and He (2017) in marginalising out the $\{X_t^i\}$ processes to get valid inference on $\boldsymbol{\beta}^i$.

If we have inferential interest in φ^i and the $\{X_t^i\}$ sequences themselves, we cannot aggregate through X, and must either solve a very high dimensional likelihood problem, or use Bayesian methods, similar to Klingenberg (2008). Bayesian methods also allow us to place a flexible hierarchical structure on our parameters. We can also make our problem more general, and have the parameter vectors vary over time.

Details of our hierarchical setup are given in appendix A.2.

6.4.5 Comparative Time Series

In theory, if our model is well fit and correct, we can solve integrals, or even simulate, to work out the unconditional effects of $\boldsymbol{U}_t^i$ on future observations, Y_τ^i, $\tau \geq t$. However, this requires our model to be well specified to get unbiased estimates, and also requires our model to be fully identified. We would like to be able to make valid inference on the unconditional effects even if the best our model can do is have good predictive properties. If our model was misspecified and could only guarantee unbiasedness of the X sequences, we would still like to get marginal inference. Further, as we will discuss in section 6.4.7, the regression does not directly measure the lottery effects on adherence.

Assume we have a dichotomous covariate, V_τ^i, which can be an element of our covariate vectors $\boldsymbol{U}_{\tau'}^i$ at some potentially different time τ'. Assume it takes values in $\{0, 1\}$. We can think of this as a treatment variable.

Our desire is to create quasi-experimental data, following Campbell and Stanley (1963). For any given time τ, we can run one of two separate procedures: (a) run our model up to time τ; (b) take the results from our model run on the full dataset at time τ. From either of these two procedures, we will get a distribution for X_τ^i, and thus a distribution for $\mathbf{P}(Y_\tau^i = 1)$.

We can separate our sequences into those with $V_\tau^i = 1$ and those with $V_\tau^i = 0$, and call these "treated" and "control" respectively. We use the model output X_τ^i values to match "treated" sequences with "control" sequences. In our data, V_τ^i is the lottery result on day $\tau - 1$. To be specific, it's an indicator variable for one of the four possible lottery results. Thus we also match sequences on $Y_{\tau-1}^i$, to make sure V is just the effect of the lottery. That is, when two people are both eligible, i.e. both adherent, but only one wins, we can match them on lotto wins; when two people are both not eligible, i.e. both not adherent, and one gets a regret message, we can match them; when one person is eligible and the other isn't, we don't match them.

From here we can apply the logic of interrupted time series: compare the subsequent sequences Y_t^i, $t \geq \tau$, for the treated and control. Since our sequences were matched at time τ, we don't have to fit time series models to the sequences: we can simply compare the differences, and conclude the marginal effect of V. This gains us both internal and external validity, since for example we don't have to worry about the effects of V potentially being different for different X levels. The downside is that we can no longer use just any control for any treated.

If V is continuous, we can still do this analysis, but we would have to change the strict definition of treated and control. One option would be to treat it as a dose-response model.

6.4.6 Matching Details

We'll make the above section more concrete. For every day t of the study, we find our set of "treated" participants for each lottery effect. That is, we find all those who won the small lottery on day t; all those who had a small regret on day t; all who had a large win on day t; all who had a large regret on day t. Our set of potential controls are those who did not have a lottery effect on day t. To measure the effect of wins, we only compare lottery winners to participants who also completed the goal on day t, but didn't win the lottery. To measure the effect of regret messages, we only compare regret message receivers to participant who also were not eligible.

We have the option to match with replacement or without, i.e. we can match each control to multiple treated, or match each control to at most one treated. Allowing matching with replacement reduces our bias since we can choose the closest possible controls for each treated (Rosenbaum 2002). The downside is potential variance increase; however in our cases, blocking multiple uses of controls reduces our sample size enough to more than cancel any variance decrease, so we match with replacement.

We determine the distance of each control to each treated participant based on $\mathbf{P}(Y_t = 1) = \Phi(X_t)$, the predicted probability of adherence on day t. We then compare $\Phi(X_t^{\text{control}})$ and

$\Phi(X_t^{\text{treated}})$, and if they're within a given tolerance δ, the control participant is matched to that treated participant. We also allow many-to-one matching: multiple controls are used if available. In typical MCMC fashion, we compute $\Phi(X)$ for every iteration (post burn in), and average through them.

We make comparisons over ten days, i.e. we record the difference in adherence between our treated participants and our controls over a ten day period. Optionally we can further restrict matches by using a cooling off period: a matched participant must wait a given number of days to be used in another comparison. If we used five days as a cooling period, a participant cannot be part of a match on day 5 and also day 7 - we'd wait until day 11 to rematch them.

These matches give us a set of comparisons for all lottery types across the timeframe of the study.

One downside of matching with replacement is that the variance calculation becomes more difficult, particularly in many-to-one matching. We overcome this by bootstrapping to get the variance instead. Details are given in appendix A.3.

6.4.7 Comparing Regression with Matching Output

One might ask why we would want matched estimates after fitting our regression model. The main reason is that the matched estimates more directly answer the question of interest: how much effect does the lottery have, and what is the mechanism? The direct estimates of the lottery effects also come with directly estimated standard errors.

The output of the regression method does not automatically inform us about the output of the matching method. For example, if a study has an extremely high adherence rate, this will limit the maximum effect size of the lottery: you can't make a discernible positive difference in a participant's adherence rate if they're already going to complete the goal 98% of the time. The autoregressive parameters also affect the propagation of the lottery effects: a participant who nearly always just repeats today what they did yesterday will

	Study Day									
	14	15	16	17	18	19	20	21	22	23
45	1	0	0	1	1	0	0	1	1	0
21	1	1	1	0	1	0	0	0	0	0
62	0	1	0	0	0	0	1	1	0	0

Table 6.1: Adherence after matching on day 13

on average respond less strongly to any incentives. Thus if we want to know the change in adherence from the lottery, we need to measure the change in adherence from the lottery.

On the other hand, the regression method doesn't suffer from these shortcomings, thus we might expect our parameters to generalize beyond the scope of the study group more readily than the in-sample lottery estimates.

6.5 Matching Example and Graphical Summaries

Given the set of matches generated according to section 6.4.6, we get a set of differences for all four lottery types. For each type, we may have multiple differences on a given study day, or none. Note that a given difference refers to one treated participant, and at least one but potentially many controls.

6.5.1 Matching Example

To give a specific example: if participant 45 wins the small lottery on day 13, we search for controls. Say that participant 45 has an expected value of $P(Y)$ of 0.6 (corresponding to $X = 0.25$[3]), and participants 21 and 62 both complete the goal on day 13 but don't win the lottery, and have expected values of $P(Y)$ of 0.55 and 0.63, within our set threshold $\delta = 0.05$, i.e. $\mid 0.6 - 0.55 \mid \leq 0.05$ and similarly for 0.63.

The adherence vectors for the three units are given in table 6.1.

With just one control, we simply subtract; with multiple controls we average the controls,

[3]We have $\mathbf{P}(Y = 1) = \Phi(X)$, but indeed $\mathbf{EP}(Y = 1) = \mathbf{E}\Phi(X) \neq \Phi(\mathbf{E}X)$. We can match on $\mathbf{E}X$ instead of $\mathbf{E}\Phi(X)$ if we want, but note that it won't be equivalent, and is generally worse in simulation

then subtract. Thus here, our difference vector would be:

$$\mathbf{Y}^{45}_{14:23} - \left(\frac{\mathbf{Y}^{21}_{14:23} + \mathbf{Y}^{62}_{14:23}}{2}\right)$$
$$=(0.5, -1, -0.5, 1, 0.5, 0, -0.5, 0.5, 1, 0)$$

Based on this comparison alone, our estimate for the one day effect of the lottery would be an increase of 0.5; our five-day estimate would be $0.5 - 1 - 0.5 + 1 + 0.5 = 0.5$, our ten-day estimate would be the sum of all ten, or 1.

Of course we don't just use one, hence our estimate for the one-day effect is the average of the first element of all such difference vectors; the five-day effect is the average of the sum of the first five element of all such vectors, and similar for the ten day.

6.5.2 Graphical Summaries

In the previous section, we work through an example of how we form the difference vectors for the matching. We can aggregate these vectors in many ways, e.g. we could stratify by different values of X, to see the effect of the lotteries separately for high-adherence and low-adherence participants.

Graphically, it's informative to view a rolling average of the effect of the lotteries over the course of the study, i.e. a smoothed estimate for every study day. Note that this is more effective for the small wins and regrets, as the large wins and regrets don't happen frequently enough to produce a smooth rolling estimate. Further, even with the small lotteries, the noise involved makes producing a useful standard deviation difficult. The appropriate amount of smoothing will depend on the size of the study.

We also produce an overall estimate of the effect of the four lotteries, for one-, five-, and ten-days. Finally, we split the study in two, and compute the estimates separately for the

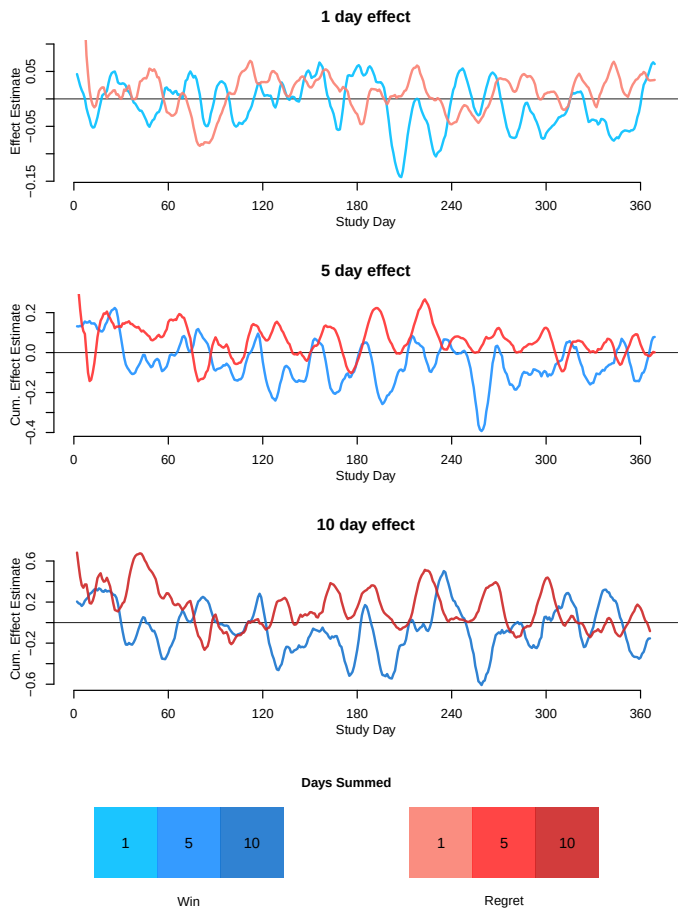


Figure 6.2: Smoothed daily estimates of small lottery effects, Shared Incentives. Each point is a moving average estimate from a five day window.

first half and second half, to evaluate any estimated change in the effect of the lottery. These estimates are given with valid (bootstrapped) confidence intervals.

6.5.3 Shared Incentives Output

The above graphical summaries are provided for the Shared Incentives data. The rolling estimates for the small lotteries are provided in Figure 6.2, and the overall and per-half estimates for small and large lotteries are provided in Figure 6.3. Blue lines and boxes describe effects for lottery wins; red lines and boxes describe effects for regret messages.

The three plots in Figure 6.2 are formed by first getting estimates for every study day

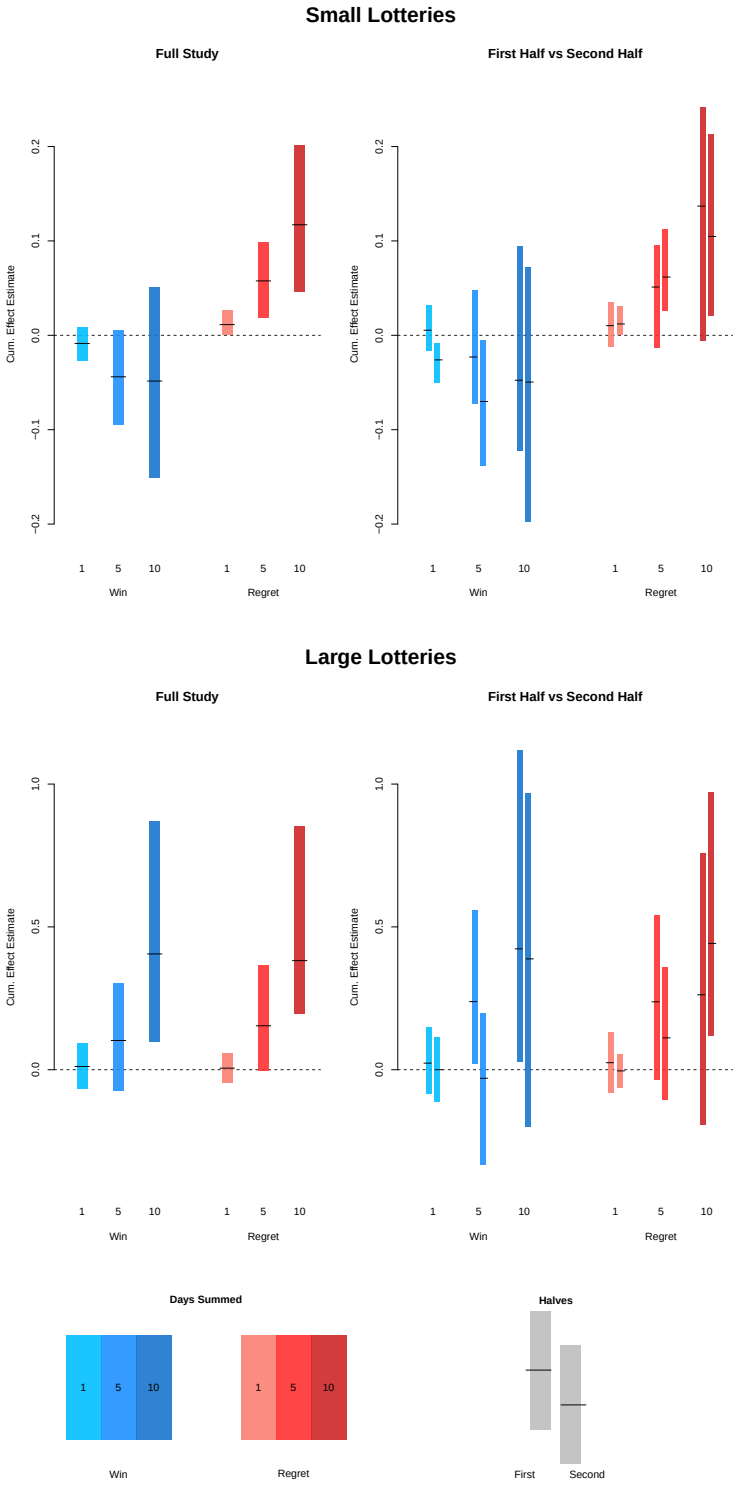


Figure 6.3: Overall and per-half estimates of all lottery effects, Shared Incentives.

through averaging, and then computing a five day rolling average. So the value at e.g. day 26 is the average of all estimated treatment effects for lottery winners on day 24, 25, 26, 27 and 28. We don't plot the daily large effects: they are sparse, and the gain from plotting the daily version seems minimal.

These plots are done separately for the one-, five- and ten-day estimates.

In Figure 6.3, we summarise the effect (i) over the course of the whole study (ii) separated by each half. These are plotted with bootstrapped confidence intervals.

There is evidence that the small regret messages are having a positive and significant effect on adherence, and both large lotteries appear to have a positive effect of an extra 0.4 adherent days in the ten day period after receiving the message.

6.5.4 HeartStrong Output

We also produce graphical summaries for the HeartStrong data. The rolling estimates for the small lotteries are provided in Figure 6.4, and the overall and per-half estimates for small and large lotteries are provided in Figure 6.5. Blue lines and boxes describe effects for lottery wins; red lines and boxes describe effects for regret messages.

We see that the regret effects have much higher variance, both at the daily level, and the overall level. This is a consequence of very high (92.5%) adherence overall, averaged over participants. Thus there are far more winning lotteries than regret lotteries, both small and large.

Overall, there appears to be little to no effect of any lottery type. This is partially due to the very high baseline adherence in this study (92.5%).

6.6 Simulation Study

We simulated datasets according to our model, with a factorial design: number of participants, using 20, 100 and 300; length of sequences per participant, using 200 and 500 days

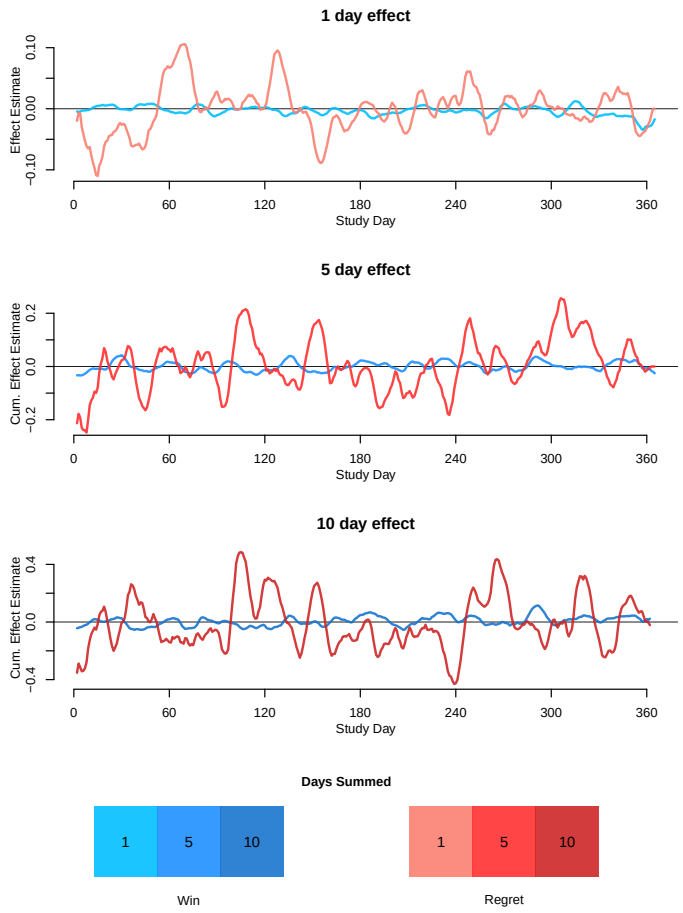


Figure 6.4: Smoothed daily estimates of small lottery effects, HeartStrong. Each point is a moving average estimate from a five day window.

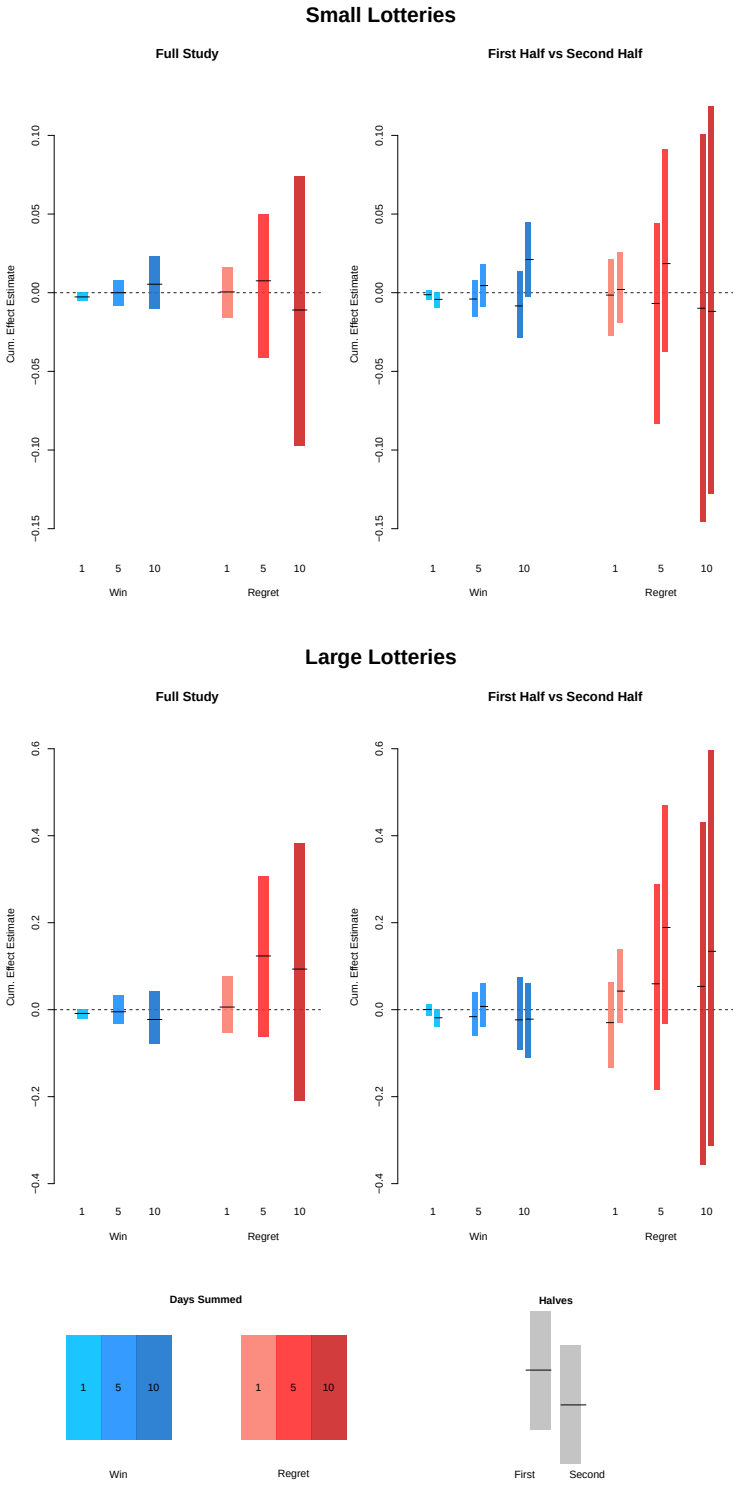


Figure 6.5: Overall and per-half estimates of all lottery effects, HeartStrong.

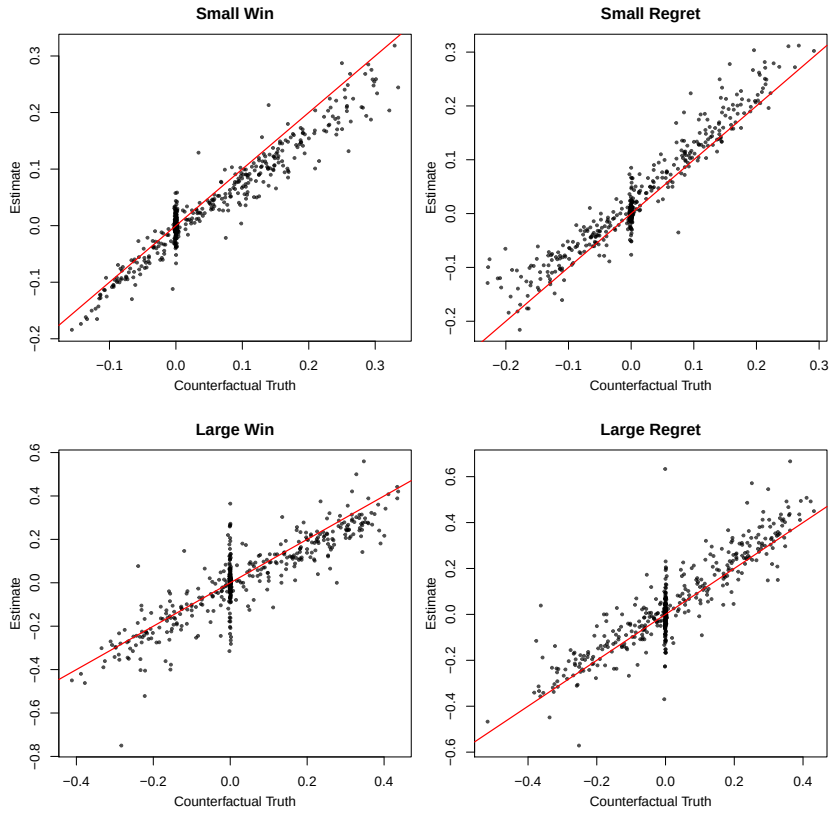


Figure 6.6: Estimated one-day lottery effects against counterfactual truth

per person; value of parameters, either all zero, or randomly nonzero.

Appendix A.1 details some aspects of this simulation. We focus on using the resulting X values to perform our matching.

For each simulation, we can work out the true value of the lottery at any time point t by simulating data up to time t, setting the lottery results on day t to take our desired value, and then simulating any number of days beyond. This is computed as a function of the parameter means and standard deviations.

For each MCMC result, we calculate the estimated lottery effects as per section 6.4.6, and bootstrap the intervals.

Figure 6.6 plots the estimated values against the counterfactual truth, for the one-day effect.

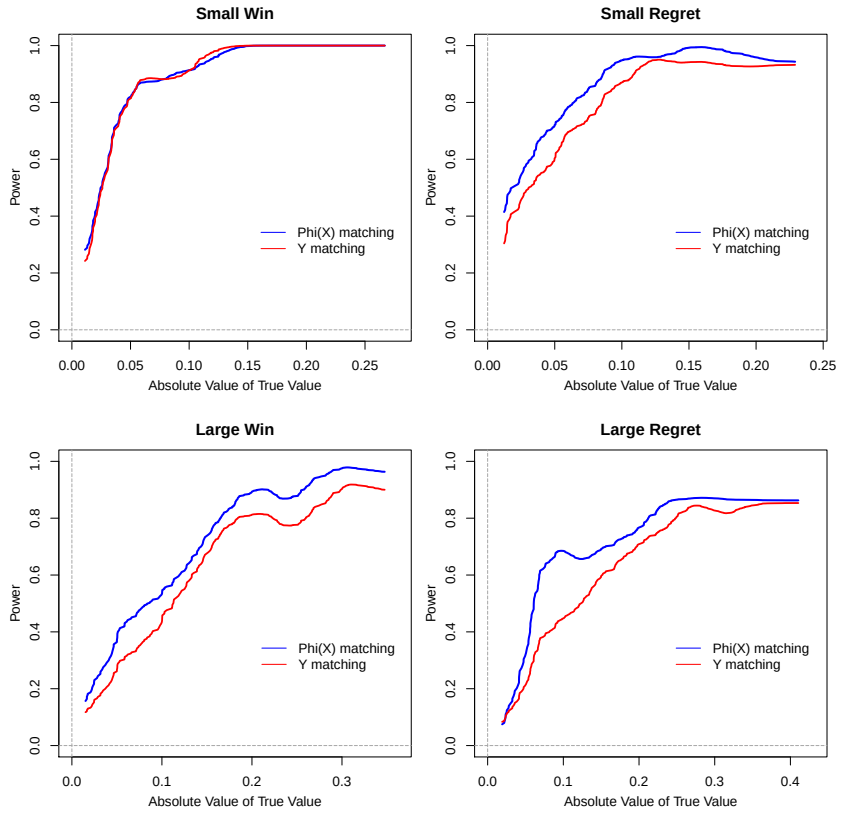


Figure 6.7: Power as a function of (absolute) counterfactual truth

Figure 6.7 plots power as a function of the absolute value of the counterfactual truth. These plots contain two lines: the blue line is for matching as we've described in this paper: run the MCMC, match on $\Phi(X)$. The red line is matching on Y itself: at time t, compute the running average of the most recent k days, and match on those. For these comparisons, we use $\delta = 0.03$, and for Y we match exactly on the mean with $k = 10$[4], which gives approximately the same number of controls for both. We see that $\Phi(X)$ matching has higher power at all levels of the true value for all lotteries except Small Wins, where the two are about equal. We will discuss the method on matching on Y further in section 6.7.4.

Coverage reaches the nominal value for null lottery effects: we get 95% coverage of zero for simulations with no lottery effect. Coverage drops to 70% in the worst cases at the extreme

[4]Matching exactly on the mean Y doesn't mean we match exactly on the last 10 days, only that two people must have completed the goal the same amount of times in the last ten days to be matched

values of the counterfactual lottery; partly this is due to correlation between the lottery coefficients and the decay coefficients.

We use the simulation study in the following section.

6.7 Sensitivity Analyses

6.7.1 Days of Matching

We somewhat arbitrarily use 1, 5 and 10 days to evaluate the effects of the lotteries. We can use any number of days, although as mentioned in section 6.3, we aren't trying to assess the total value of the lottery with this analysis, but specifically the shorter term effects.

In Figure 6.8, we plot the counterfactual truth for cumulative effects from one day to ten days using our simulated data; we plot non-null cases. The top plot is small wins and the bottom is large wins. We see that small wins are almost all flat after six days, while large wins are just starting to flatten at ten days.

We extend this in Figure 6.9, where we plot the analogous estimated effects, going to twenty five days. Again, we see very little difference beyond ten days.

In applied work, we can of course estimate any number of cumulative days.

6.7.2 Control Eligibility: Matching on Propensity

In order to be a valid match, we require a control to be within a given tolerance, as outlined in section 6.4.6. If the tolerance is large, we get more matches but less quality matches. If the tolerance is small, the matches are quality, but the number of matches could be low.

In Figure 6.10, we plot the distribution of the $\Phi(X)$ values from the Shared Incentives study as a histogram of all values, and in Figure 6.11, we plot curves representing a participant's expected value of $\Phi(X)$ for every day of the study; one third of the curves are given. While it looks messy, this implies matching is achievable most days for most participants, as we

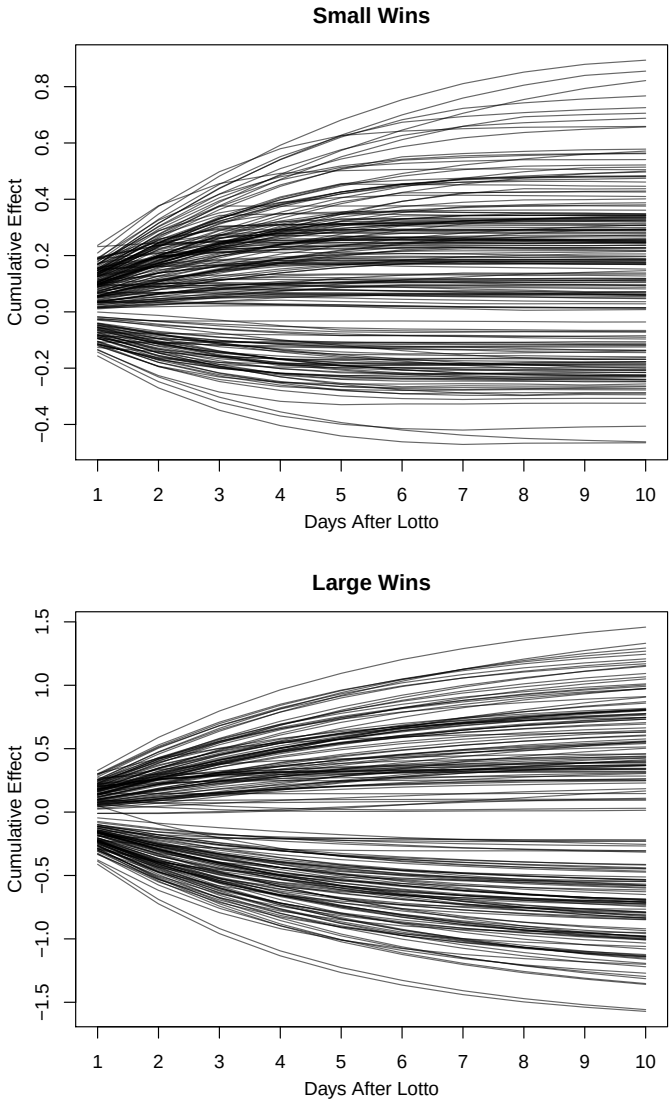


Figure 6.8: Counterfactual true cumulative lottery effects from one to ten days

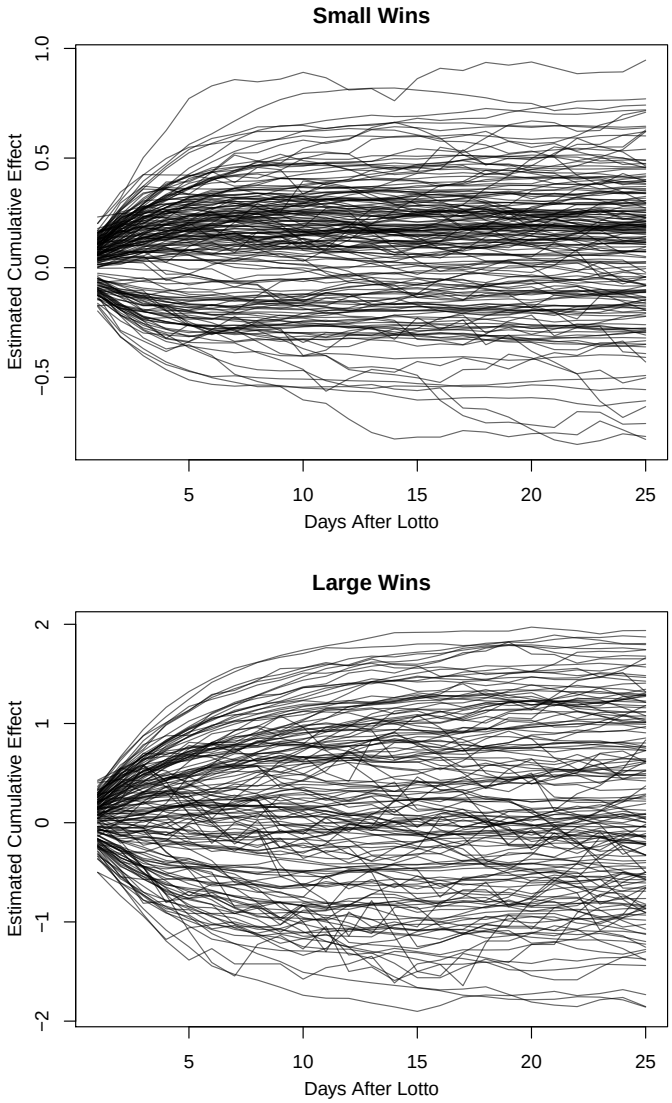


Figure 6.9: Estimated cumulative lottery effects from one to twenty-five days

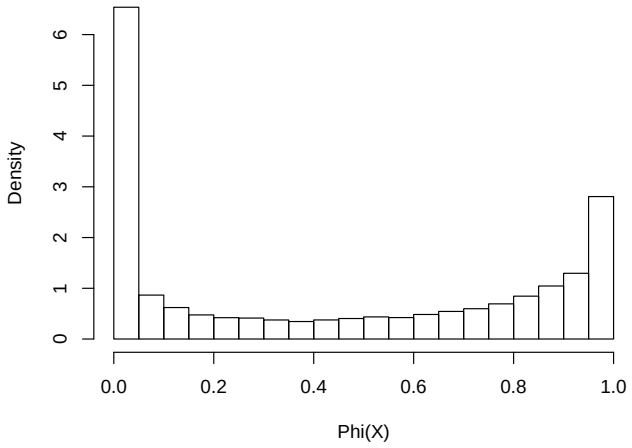


Figure 6.10: Shared Incentives: Histogram of $\Phi(X) = \Pr(Y = 1)$ for all participants and days

have a lot of overlap.

Further, in Figure 6.12 we plot the mean squared errors of the estimates as a function of the tolerance δ, averaged over our simulation results. This plot is for the one day effect. Note that for these results, we allow at most three controls for each treated, so as $\delta \to 1$, we don't tend towards allowing random matches, but instead the best three are chosen if all controls are available, hence why the error doesn't blow up.

The best δ depends on how many participants you have, and to a lesser degree, how many days in each series. The small and large lotteries will have different optimal solutions, as will the one-, five- and ten-day effects.

For real data, we cannot assess the bias in the bias-variance tradeoff separately. Instead, we can plot the number of formed matches as a function of δ. This serves as a good proxy for variance, especially if you use a bound on the number of controls for each treated.

Note that our priority is not out of sample prediction; reducing squared error is not our primary goal. Bias is generally a more significant concern than variance, as a bias error is

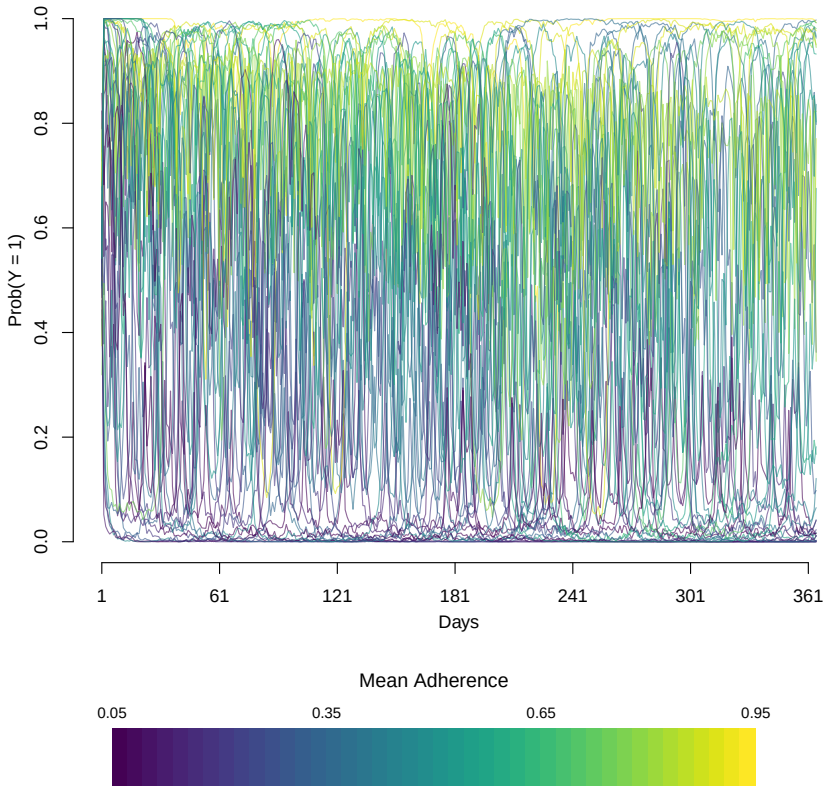


Figure 6.11: Shared Incentives: Plot of $\Phi(X) = \Pr(Y = 1)$ curves; each curve represents one participant

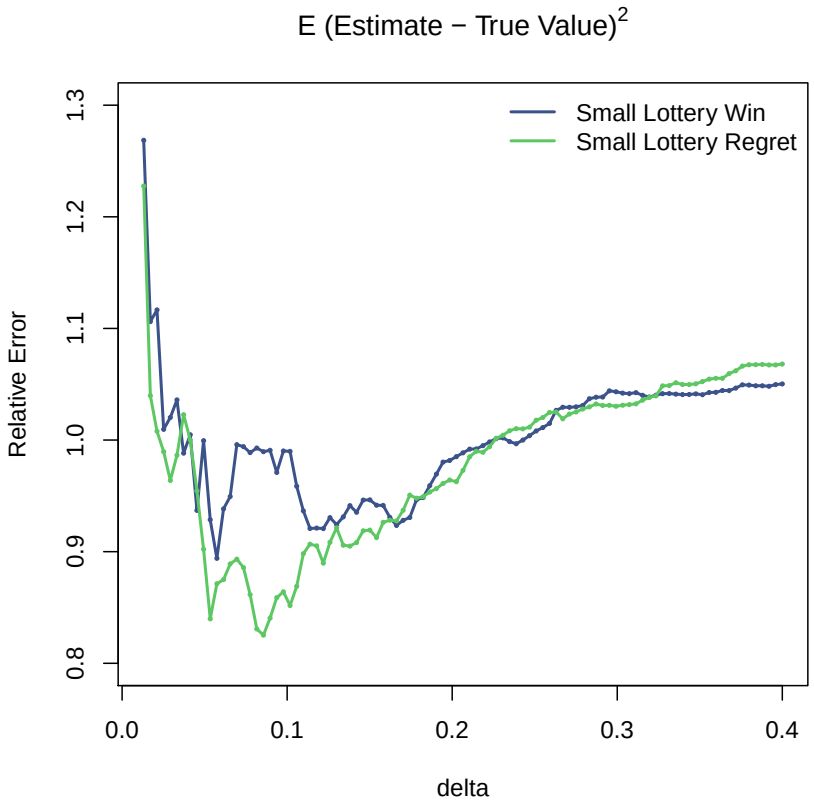


Figure 6.12: Simulation Study: Plots of mean squared error for the small lottery, for one-day estimates

hidden from us; we tend to favor small values of δ. We still care about power, so variance is certainly not irrelevant.

6.7.3 Control Eligibility: Multiple Matches

In our match, we allowed each participant to be involved in multiple comparisons starting on a given study-day. We also don't restrict concurrent matching: we allow for example a participant to be a control starting on day 11 and then a treated unit starting day 13.

Avoiding this provides restrictions on the number of matches we could create, with the benefit of avoiding correlated differences if a participant is e.g. used as a control from day 11 till 20, and a treated from day 14 to 23. Depending on your desired δ and number of participants, avoiding concurrent matches could have a significant impact on the total number of matches. Here we evaluate the effect of changing this restriction.

We can increase the variance but potentially decrease the bias by tightening this restriction. We can allow a participant to be freed to be matched after a fixed number of days, e.g. five, or as mentioned go all the way up to a ten-day restriction, which is equivalent to not allowing a person to be involved in more than one match simultaneously.

Our desire to avoid bias depending on our period of interest. If our one- and five- day effects are of primary interest, a ten-day restriction will lose us matches that might not have caused us concern. However, if the ten-day effect is of primary interest, we may be more inclined to keep the restriction closer to ten days, assuming we can justify the restriction with a large sample size.

Figure 6.13 plots the standard errors of the estimate for small wins for one-day, comparing matching with and without replacement when the true effect is zero. Note that coverage for the duplicates matches is actually slightly higher than the non-duplicated matches. The relationship between the two is the same for non-null effects.

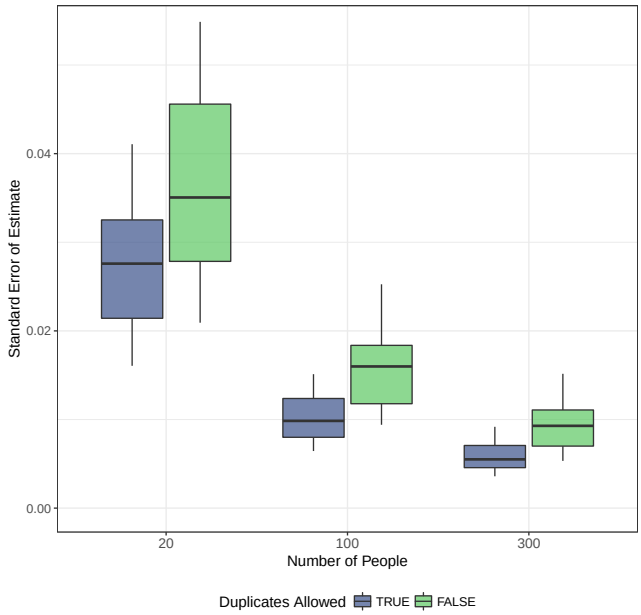


Figure 6.13: Standard errors of the estimates for small wins, one-day, for null effects.

6.7.4 Matching on Adherence

Instead of running our MCMC and matching based on the output, we could instead simply match directly on the $\{Y_t\}^i$ sequences: for a given day τ, we can match participants based on some function of days up to and including day τ. For example, we could match on an average of the most recent ten days, or a weighted average with the most recent days weighted more heavily; we could match exactly on a small sequence of days; we could match both means and number of switches[5] of the most recent twenty days.

Such a procedure avoids the entire problem of fitting the MCMC. Of course we may be interested in the outcome of the model too, but for now we'll focus just on the matching results.

In section 6.6, we compared matching on $\Phi(X)$ to matching on the most recent ten days;

[5] A switch is going from adherent to non-adherent on consecutive days, or vice versa. The sequence 010101 has five switches, while 000111 only has one

i.e. we can match participant 13 and participant 94 on day 17 if:

$$\frac{1}{10}\sum_{t=8}^{17} Y_t^{13} = \frac{1}{10}\sum_{t=8}^{17} Y_t^{94} \tag{6.8}$$

In our simulations, this procedure results in lower power than $\Phi(X)$ matching. In our simulations, this was due to the variance of the estimates from matching on Y, measuring from the counterfactual truth, were higher for all lotteries and cumulative estimates. We don't recommend this procedure as a final outcome, however is it a useful sanity check on the outcome of the MCMC procedure.

6.8 Discussion

In this paper, we propose a two-pronged approach to analyzing binary time series with intermediate outcomes. First, we fit a parameter-driven regression model with a latent process; second, we use the outcome of that model to form matches and estimate effects.

We can think of this whole process as an extended propensity match. We first build and fit a latent model: this gives us probabilities for each participant, each day. Secondly, we then use this to form matches, based on minimising probability distance: this is exactly the concept behind propensity matching.

Once we form these matches, we break the time series up into segments: a time series interrupted by the lottery results. These matched segments form our lottery estimates, and thus we have our estimate effects on the correct scale.

In our studies, we saw a stronger effect in both cases from the regret messaging. This backs up much of what regret theory indicates: we hate to lose more than we love to win.

It is hard to judge in our studies if the large lottery is sensible, in terms of providing ten times the effect of the small lottery. In both studies the large lottery effects do seem larger than the small lottery effects, but the results are consistent with many possible patterns of effect. We could read these results as a validation of having a large and small lottery;

alternatively it's possible the large dampens the effect of the small. We also must remark that it appears second half effects are not overly damped compared to the first half: it is encouraging that the study matters to participants more than six months into it.

A more general question is whether the lottery is sensible at all: the strongest effects measured are 0.5 extra goal completions days in the ten day period following the lottery. As discussed in section 6.3, this "interior" analysis of the lottery is not the best way to answer the question of the lottery's total worth however.

Chapter 7

Discussion

Current matching methods have already been put to great use. We offer a new methodology to make them even better. Wherever possible, we make explicit what other methods make implicit; we search for the best parameters where other methods require a user choice. Most importantly, we offer a framework for evaluating choices and parameters in a matching context.

We've detailed two categories of problems for which matching offers a reasonable solution: a multidimensional observational study of a fixed environment, Philadelphia, and how factors of the environment affect how real people operate in it.

Secondly, we take a set of controlled experiments with uncontrolled interiors, i.e. the moving parts of the experiment are not controlled. We develop methods that allow us to recreate a form of propensity matching on this interior, to learn about the dynamics of the treatment effect: the lotteries and corresponding messaging system.

For our matching methods, we always want to take the best of all possible worlds. So when evaluating a match, we want to make sure we've organised our match in such a way that no methods we can think of can learn enough from a subset of our formed pairs to be able to tell apart the rest of the formed pairs, in terms of guessing treatment or control.

There is no restriction on how we form a match: if we were so inclined, we could even build a match by hand. But if that match is not balanced, so much so that a powerful algorithm can predict which unit is treated, we've done a poor job. We use this to our advantage: generate as many matches as our computers can handle, and select based on predictability. When that's not enough to tell the matches apart, we can directly check covariate balance.

Urban Analyses It seems safe to conclude that crime mostly happens close to businesses. Even when all other aspects are as close as we can make them, people are simply more likely to be found near businesses, and crime happens near people. Slightly unsatisfactorily, the results of both chapters 3 and 5 leave us wanting more data, and especially more data about people.

We hope that any researchers with such data can build on what we've done: using our methods, our code or even our data to better understand the complicated and beautiful world of urban environments.

As we discuss in section 5.4, it's hard to use these analyses to come to policy conclusions. We hope at the least we've made it easier for future researchers to use our methods, data and tools to make progress in this area.

As linked to in the introduction, the data from the core analyses is currently available at `https://github.com/ColmanHumphrey/urbananalytics`. From the same github page, we will soon make available all data[1] and tools used to generate all work in this thesis.

Lottery Analysis We direct the reader to the discussion at the end of the previous chapter, in section 6.8.

[1] We might not be able to release the business data due to use agreements - in review.

Appendix

A.1 Simulation Study

Each simulation was run with 2,000 MCMC iterations. We plot four typical runs in figure A.1, with horizontal lines for the true parameter values; convergence is very fast, from a random start.

Figure A.2 plots power against absolute value of the true parameter.

Coverage at zero is at the nominal value for all six parameters, however coverage drops to about 80% away from zero. True coverage is regained if a bootstrap method is implemented: bootstrap at the level of participants and run the MCMC algorithm on each bootstrap sample. This can be an expensive step in terms of computation.

The φ parameters have one issue near zero, in that the estimate won't be negative. This by itself is not a problem, but the variance of the overall mean is correlated with the φ parameter, and it's worse the closer φ is to zero. Figure A.3 plots the estimated values against the simulated truth, with the points colored according to the standard deviation of α.

The decay parameters are poorly identified, but the log-likelihood is significantly higher with them than without. The main issue is the parameters are highly correlated: both the decay and the "length" parameters are correlated, and both are correlated with the respective lottery coefficients.

A.2 Hierarchical Structure and Gibbs Sampling

For the majority of our parameters, we have a basic hierarchical structure of the form:

$$\beta_j \sim \mathcal{N}(\beta, \sigma^2) \tag{A.1}$$

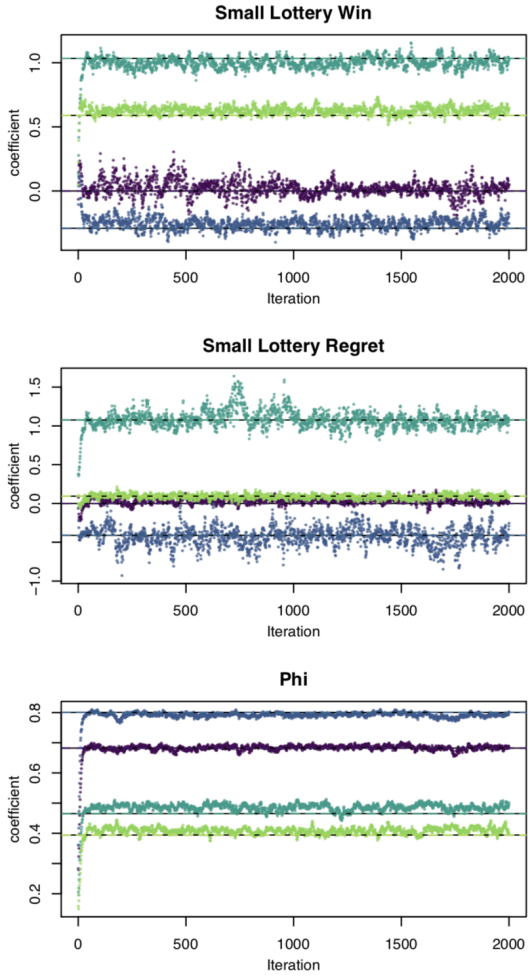


Figure A.1: MCMC iterations for the small lottery and φ coefficients

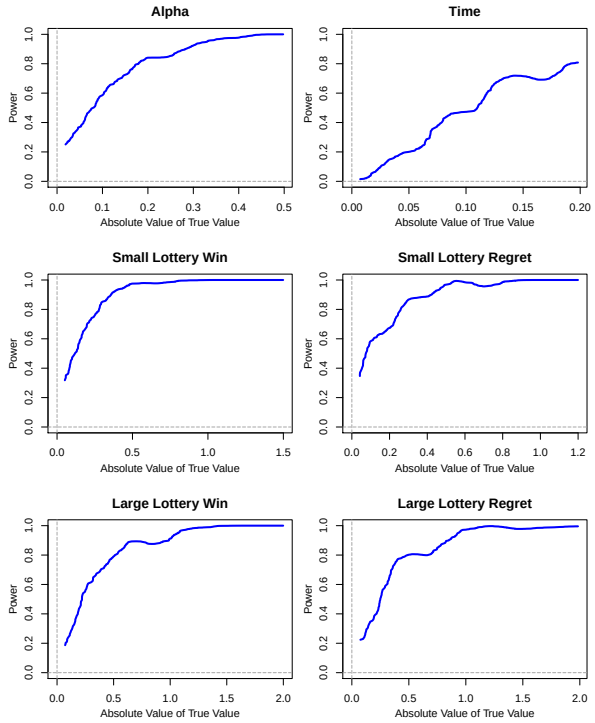


Figure A.2: Power calculations for our main six variables

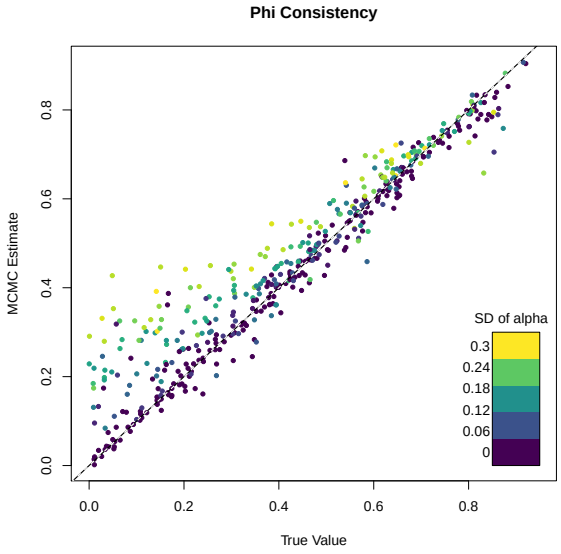


Figure A.3: Estimated φ against ground truth; color according to the standard deviation of the overall mean.

Where β is the parameter of interest, and β_j is the value of that parameter for participant j. β itself is assumed to come from a very flat distribution with mean zero, and the variance is fit straight from lme4 random effects models. The user can adjust the functions for greater flexibility.

We use this structure on the intercept coefficients, the time coefficients, the small win lottery coefficients, the small regret lottery coefficients, and the φ parameters. The exceptions are the large lotteries. Under simulation, we require huge amounts of data to fit individual large lottery parameters to each participant. This is unsurprising, since the average participant only wins a large lottery 3.5 times, let alone a further split into winning and regret. If your data can support individual large coefficients, then of course you can fit them.

Note that from simulations, the lme4 defaults perform very well when compared with pure Bayesian methods, and for the variance parameter, lme4 behaves similarly to flat priors with a spike at zero. lme4 models will also "blow up" when the data cannot support the model, a feature hidden in the form of a bug. This information is what leads us to avoiding fitting individual parameters for the large lotteries.

For the Gibbs sampling, we sample from our generated lme4 models.

A.2.1 Hard to Fit Data

If the overall adherence is extremely high, as it is in the HeartStrong study, or extremely low, the parameters will not be strongly identified. Similarly we will have problems if the number of switches in the binary sequences is extremely low.

For the HeartStrong study, this was overcome by having no time coefficient, and only two hierarchical parameters: the intercepts for each participant and the autoregressive parameters. With these restrictions, the MCMC converges very quickly.

A.3 Bootstrapped Intervals and p-values

In standard Bootstrapping with replacement, we resample our data multiple times, compute our estimates and potentially standard deviations for each resample, and combine those with the estimates on the full sample to produce inferences. Applying this, when we're computing our matched differences we could sample each treatment-control combination, potentially many-to-one matching, with replacement. But this would not correctly estimate the variability of our data: if each participant has slightly different lottery behaviour, then this method will underestimate the noise of each resample.

Similar to using random effects, we must be slightly more careful when resampling with our data: we should sample *participants* with replacement. This can happen at any stage of inference: we can resample participants when we are calculating our difference estimates, or we could do it from the very start and run the entire MCMC analysis on resampled datasets. Note that this would be expensive.

It is non-trivial to generate confidence intervals and p-values from bootstrap estimates. We follow Hesterberg (2015) and use our bootstrap samples to approximate the distribution of the t values produced by differences.

We're interested in some covariate β. We have the estimate $\hat{\beta}$, and the standard error $\hat{se}_\beta$ from our base estimation. In testing against zero, we generate:

$$t = \frac{\hat{\beta} - 0}{\hat{se}_\beta} \tag{A.2}$$

We might not believe that this t statistic will have a t distribution, due to e.g. lack of independence. Instead of assuming a distribution on this statistic, we use our bootstrap samples to estimate the distribution: we generate R bootstrap t statistics in the same way

Table A.1: Empirical Distribution After Matching for Pairs

$\mathbf{x}_t$	$\mathbf{x}_c$	Proportion
(0, 0)	(2, 0)	1/6
(1, 0)	(0, 0)	1/6
(2, 0)	(1, 0)	1/6
(0, 1)	(1, 1)	1/6
(1, 1)	(2, 1)	1/6
(2, 1)	(0, 1)	1/6

as we do in the base estimation:

$$t^* = \frac{\hat{\beta}^* - \hat{\beta}}{\hat{se}_{\beta^*}} \tag{A.3}$$

Letting q_α be the α quantile of the bootstrap t distribution, the interval is:

$$(\hat{\beta} - q_{1-\alpha/2}\hat{se}_\beta,\ \hat{\beta} - q_{\alpha/2}\hat{se}_\beta) \tag{A.4}$$

Just like with regular confidence intervals, we can reject zero at level α. We can use this to form p-values: we calculate the largest α such that this interval contains zero.

Hesterberg recommends $R > 15,000$ for published works, and we agree.

A.4 Pair Differences

Say the covariate vector was two-dimensional, with the first element uniform over $\{0, 1, 2\}$, and the second independently uniform over $\{0, 1\}$. Say our match procedure produced the distribution for pairs given in table A.1.

We can then look at the distribution of $\mathbf{x}$ for treated versus control, given in table A.2. This is formed from table A.1 by calculating the proportion a given set of characteristics shows up as a treated unit, and take the ratio of that relative to the proportion it shows up at all. For example, for units with covariates $(0, 1)$, we find this pair shows up as 1/12 of all pairs

Table A.2: Empirical Distribution After Matching, Unpaired

$\mathbf{x}$	Proportion Treated
(0, 0)	1/2
(1, 0)	1/2
(2, 0)	1/2
(0, 1)	1/2
(1, 1)	1/2
(2, 1)	1/2

Table A.3: Empirical Distribution of Differences After Matching

$\mathbf{x}_i - \mathbf{x}_j$	Proportion i Treated
(2, 0)	1/2
(-1, 0)	1/2
(1, 0)	1/2
(-2, 0)	1/2

as a treated unit, and 1/12 of all pairs as a control unit, thus half the time we see it it's a treated unit.

We can also look at the distribution of differences, in table A.3. The formation of this is slightly different: we imagine we look at pairs in random order, compute the difference, and then look at which unit was treated. For example, the difference $(-2, 0)$ shows up when we pick up the pair $\big((0, 0), (2, 0)\big)$, the first element of table A.1, and also when we pick up the pair $\big((0, 1), (2, 1)\big)$, which is the last element of table A.1 picked up "backwards". Exactly half of such pairs have the first unit treated, and the other half have the second treated, thus we get 1/2.

This almost sounds trivial: if table A.1 had no last element, we'd still see both $(-2, 0)$ and $(2, 0)$ half the time. But the sign of these differences would be perfectly correlated with the outcome: every time we saw $\mathbf{x}_i - \mathbf{x}_j = (-2, 0)$, we'd know unit i was the treated unit, not unit j. Thus the differences would not be symmetric conditioned on treatment.

Given these resulting distributions, any analysis on the units directly would have zero predictive power, and any analysis on the differences would have zero predictive power.

However, the joint distribution gives the whole game away: we can perfectly predict which unit is treated given $(\mathbf{x}_i, \mathbf{x}_j)$. For example, if we see a pair with covariates $((2,0),(0,0))$, we predict the second unit will be treated, since that's the only ordering possible from table A.1.

Bibliography

References for Chapter 1

Heckman, James J, Hidehiko Ichimura, and Petra Todd (1998). "Matching as an econometric evaluation estimator". *The review of economic studies* 65.2, pp. 261–294.

Kolmogoroff, Alexander (1933). "Grundbegriffe der wahrscheinlichkeitsrechnung".

Pearl, Judea et al. (2009). "Causal inference in statistics: An overview". *Statistics surveys* 3, pp. 96–146.

Rosenbaum, Paul R and Jeffrey H Silber (2001). "Matching and thick description in an observational study of mortality after surgery". *Biostatistics* 2.2, pp. 217–232.

References for Chapter 2

Couture, Victor and Jessie Handbury (2015). "Urban revival in America, 2000 to 2010". *American Economic Association annual meeting*. Vol. 3, pp. 24–27.

Deutsch, William (2016). "Crime Prevention Through Environmental Design". *The Balance*.

Ellen, Ingrid Gould, Keren Mertens Horn, and Davin Reed (2017). "Has Falling Crime Invited Gentrification?"

Jacobs, Jane (1961). *The Death and Life of Great American Cities*. Random House, New York.

MacDonald, John (2015). "Community design and crime: the impact of housing and the built environment". *Crime and justice* 44.1, pp. 333–383.

Simmel, Georg (2011). *Georg Simmel on individuality and social forms*. University of Chicago Press.

Verbrugge, Lois M and Ralph B Taylor (1980). "Consequences of population density and size". *Urban Affairs Quarterly* 16.2, pp. 135–160.

Winsborough, Halliman H (1965). "The social consequences of high population density". *Law and Contemporary problems* 30.1, pp. 120–126.

References for Chapter 3

Branas, Charles C et al. (2011). "A difference-in-differences analysis of health, safety, and greening vacant urban space". *American journal of epidemiology* 174.11, pp. 1296–1306.

Deutsch, William (2016). "Crime Prevention Through Environmental Design". *The Balance*.

Goodyear, Sarah (2012). "Walk Score Is Great, But it Still Doesn't Capture Walk Appeal". *The Atlantic Citylab*.

Huber, Peter J (2011). "Robust statistics". *International Encyclopedia of Statistical Science*. Springer, pp. 1248–1251.

Jacobs, Jane (1961). *The Death and Life of Great American Cities*. Random House, New York.

Simmel, Georg (2011). *Georg Simmel on individuality and social forms*. University of Chicago Press.

Team, R Core et al. (2013). "R: A language and environment for statistical computing".

Verbrugge, Lois M and Ralph B Taylor (1980). "Consequences of population density and size". *Urban Affairs Quarterly* 16.2, pp. 135–160.

Walker, Kyle (2018). *tidycensus: Load US Census Boundary and Attribute Data as 'tidyverse' and 'sf'-Ready Data Frames*. R package version 0.4.1. URL: `https://CRAN.R-project.org/package=tidycensus`.

Weisburd, David (2015). "The law of crime concentration and the criminology of place". *Criminology* 53.2, pp. 133–157.

Wickham, Hadley et al. (2017). *dplyr: A Grammar of Data Manipulation*. R package version 0.7.4. URL: `https://CRAN.R-project.org/package=dplyr`.

References for Chapter 4

Abadie, Alberto and Guido W Imbens (2006). "Large sample properties of matching estimators for average treatment effects". *econometrica* 74.1, pp. 235–267.

– (2016). "Matching on the estimated propensity score". *Econometrica* 84.2, pp. 781–807.

Bernstein, Serge (1927). "Sur l'extension du théorème limite du calcul des probabilités aux sommes de quantités dépendantes". *Mathematische Annalen* 97.1, pp. 1–59.

Bonferroni, C (1936). "Teoria statistica delle classi e calcolo delle probabilita". *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze* 8, pp. 3–62.

Buja, Andreas, Werner Stuetzle, and Yi Shen (2005). "Loss functions for binary class probability estimation and classification: Structure and applications". *Working draft, November* 3.

Chen, Hao and Dylan S Small (2016). "New multivariate tests for assessing covariate balance in matched observational studies". *arXiv preprint arXiv:1609.03686*.

Chen, Tianqi and Carlos Guestrin (2016). "Xgboost: A scalable tree boosting system". *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM, pp. 785–794.

Gagnon-Bartsch, Johann and Yotam Shem-Tov (2016). "The Classification Permutation Test: A Nonparametric Test for Equality of Multivariate Distributions". *arXiv preprint arXiv:1611.06408*.

Gu, Xing Sam and Paul R Rosenbaum (1993). "Comparison of multivariate matching methods: Structures, distances, and algorithms". *Journal of Computational and Graphical Statistics* 2.4, pp. 405–420.

Heller, Ruth, Paul R Rosenbaum, and Dylan S Small (2010). "Using the cross-match test to appraise covariate balance in matched pairs". *The American Statistician* 64.4, pp. 299–309.

Hill, Jennifer and Jerome P Reiter (2006). "Interval estimation for treatment effects using propensity score matching". *Statistics in Medicine* 25.13, pp. 2230–2256.

Ho, Daniel E et al. (2007). "Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference". *Political analysis* 15.3, pp. 199–236.

Hölder, Otto (1889). "Ueber einen Mittelwerthabsatz". *Nachrichten von der Königl. Gesellschaft der Wissenschaften und der Georg-Augusts-Universität zu Göttingen* 1889, pp. 38–47.

King, Gary and Richard Nielsen (2016). "Why propensity scores should not be used for matching".

King, Gary, Richard Nielsen, et al. (2011). "Comparative effectiveness of matching methods for causal inference". *Unpublished manuscript* 15.
Lee, Brian K, Justin Lessler, and Elizabeth A Stuart (2010). "Improving propensity score weighting using machine learning". *Statistics in medicine* 29.3, pp. 337–346.
Pearl, Judea et al. (2009). "Causal inference in statistics: An overview". *Statistics surveys* 3, pp. 96–146.
Rosenbaum, Paul R (2002). "Observational studies". *Observational studies*. Springer, pp. 1–17.
– (2005). "An exact distribution-free test comparing two multivariate distributions based on adjacency". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.4, pp. 515–530.
– (2012). "Optimal matching of an optimally chosen subset in observational studies". *Journal of Computational and Graphical Statistics* 21.1, pp. 57–71.
Rosenbaum, Paul R and Donald B Rubin (1983). "The central role of the propensity score in observational studies for causal effects". *Biometrika* 70.1, pp. 41–55.
Schafer, Joseph L and Joseph Kang (2008). "Average causal effects from nonrandomized studies: a practical guide and simulated example." *Psychological methods* 13.4, p. 279.
Sekhon, Jasjeet S (2011). "Multivariate and propensity score matching software with automated balance optimization: the matching package for R".

References for Chapter 5

Pimentel, Samuel D, Dylan S Small, and Paul R Rosenbaum (2016). "Constructed second control groups and attenuation of unmeasured biases". *Journal of the American Statistical Association* 111.515, pp. 1157–1167.

References for Chapter 6

Asch, David A et al. (2015). "Effect of financial incentives to physicians, patients, or both on lipid levels: a randomized clinical trial". *Jama* 314.18, pp. 1926–1935.
Campbell, Donald T and Julian C Stanley (1963). "Experimental and quasi-experimental designs for research". *Handbook of research on teaching. Chicago, IL: Rand McNally.*
Cox, David R et al. (1981). "Statistical analysis of time series: Some recent developments [with discussion and reply]". *Scandinavian Journal of Statistics*, pp. 93–115.
Dunsmuir, William TM, David J Scott, et al. (2015). "The glarma package for observation driven time series regression of counts". *Journal of Statistical Software* 67.7, pp. 1–36.
Dunsmuir, William and Jieyi He (2017). "Marginal Estimation of Parameter Driven Binomial Time Series Models". *Journal of Time Series Analysis* 38.1, pp. 120–144.
Klingenberg, Bernhard (2008). "Regression models for binary time series with gaps". *Computational Statistics & Data Analysis* 52.8, pp. 4076–4090.
Loomes, Graham and Robert Sugden (1982). "Regret theory: An alternative theory of rational choice under uncertainty". *The economic journal* 92.368, pp. 805–824.
Patel, Mitesh S et al. (2016). "Framing financial incentives to increase physical activity among overweight and obese adults: a randomized, controlled trial". *Annals of internal medicine* 164.6, pp. 385–394.

Rosenbaum, Paul R (1984). "The consequences of adjustment for a concomitant variable that has been affected by the treatment". *Journal of the Royal Statistical Society. Series A (General)*, pp. 656–666.

– (2002). "Observational studies". *Observational studies*. Springer, pp. 1–17.

Troxel, Andrea B et al. (2016). "Rationale and design of a randomized trial of automated hovering for post–myocardial infarction patients: The HeartStrong program". *American heart journal* 179, pp. 166–174.

Tversky, Amos and Daniel Kahneman (1991). "Loss aversion in riskless choice: A reference-dependent model". *The quarterly journal of economics* 106.4, pp. 1039–1061.

Wu, Rongning and Yunwei Cui (2014). "A Parameter-Driven Logit Regression Model for Binary Time Series". *Journal of Time Series Analysis* 35.5, pp. 462–477.